May 1978

VOLUME II, NO. 5

page 38

page 58

page 127

Cover illustration by Thomas O. Miller

Cover Story

LAUNCHING PAD

DIGGING IN

ON THE LIGHTER SIDE

FUTURE COMPUTING

DEPARTMENTS

Personal Computing

MAY 1978 VOL. II, NO. 5

Publisher/Editor
Harold G. Buchbinder

Assistant Editors
Gary Dozier
Harry Shershow
Ellen Stein
Don Wood

Art Director
Jane Higgins

Editorial Assistant
Marjorie Morse

Art Staff
Rene Stawicki
Michael Barisano
Stephen Fischer
Barbara Leonard
Richard Sarno

Editorial Production
Sarah Jewler

Advertising Production
Joe Ingram

Composition
Peter Bochner
Donna St. Cyr
Julie Ward
Katherine Willis

Circulation
Regina Harrington
Dave Harriman
Joe Gillis

General Administration
Jonna Yager
Esther Shershow
Marion Pearlman
Sarah Binder
Don Schaaf

Advertising Sales. Northeast: George Palken, Bill Mahan, 1050 Commonwealth Ave., Boston, MA 02215; (617) 232-5470 • **Mid-Atlantic:** Arthur Daks, Benwill Publishing Corp., 69 So. Central Ave., Valley Stream, NY 11580; (516) 872-9550 • **Mid-West:** Hank Bean, 2633 Hillside Lane, Evanston, IL 60201; (312) 475-7173 • **Northwest:** Ralph Petersen, 1380 Oak Creek Drive, Palo Alto, CA 94304; (415) 328-4392 • **Southwest:** Yuri Spiro, Carol Stagg, Benwill Publishing Corp., 924 Westwood Blvd., Los Angeles, CA 90024; (214) 478-3017 • **Japan:** Hiro H. Irie, International Business Corp., 11-8, Narita-Higashi 1-chome, Suginami-ku, Tokyo 166; Phone (03) 445-6111.

Subscription rates. U.S.: 1 year (12 issues) $14; 2 years (24 issues) $26; 3 years (36 issues) $38. Canada & Mexico: add $4/year for surface mail, $8/year for airmail. In Japan/Asia: Personal Computing, c/o CQ Publishing Co., 14-2 Sugamo 1-chome, Toshima-ku, Tokyo 170, Japan. All other countries: add $8/year for surface mail, $36/year for airmail. Send subscription orders and changes of address to: Circulation, Personal Computing, 1050 Commonwealth Ave., Boston, MA 02215. **Back issues.** U.S.: $3. Canada & Mexico: $4. All other countries: $6. (Please remit in US funds – Thank you)

Publisher. Published monthly by Benwill Publishing Corp.: Harold G. Buchbinder, Chairman of the Board; George Palken, President; Esther Shershow, Treasurer. Executive, Editorial and Subscription Offices: 1050 Commonwealth Ave., Boston, MA 02215. Controlled Circulation postage paid at Long Prairie, MN. Membership in Audit Bureau of Circulation pending.

Pet Dealings

Dear Editors:

I see that the January 1978 issue of *Personal Computing* has, again, devoted several pages, namely 35 & 36, to coverage of the PET computer. As a manufacturer of home computer systems, I am deeply disturbed by the press that both you and other personal computer magazine publishers are giving to the Commodore PET system.

Technico had the pleasure of being opposite the Commodore PET booth at the First West Coast Computer Faire in March 1977 when the system was introduced at $495. In a phone conversation on 1/16/78 with the Computer Factory, a PET dealer, I learned that Commodore is not delivering the 4K system advertised at $595 to their dealers and will only deliver the 8K, $795 model.

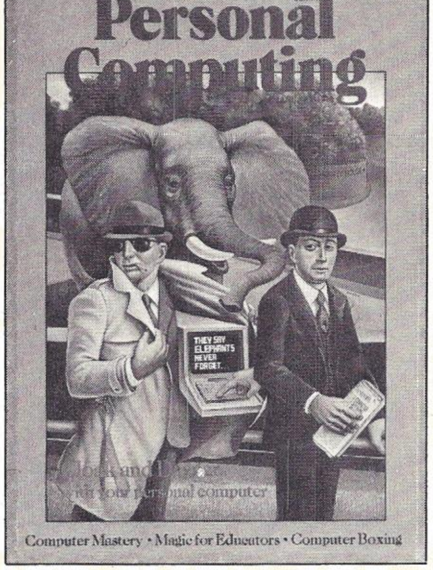
According to the Computer Factory, the 4K version, first introduced last March, is now scheduled for delivery in late spring or summer of 1978 and will carry a price tag in excess of its advertised $595 price.

Also according to the Computer Factory, the only difference between the 8K and 4K versions are the memory chips themselves.

If a company can deliver an 8K memory system, there is no reason in the world that they cannot deliver a 4K system. Since there are approximately 8 memory chips required to expand a system from 4K to 8K, and since a current selling price for 4K x 1 memory chips in volume is about $2.50, the additional 4K memory would not exceed a cost of approximately $20.00. This does not justify the $200 difference in price between the $595 and $795 system now being delivered.

We can only conclude, therefore, that the PET system advertised at $595 is a classic example of "bait & switch" whereby unsuspecting buyers are lured in by a low cost product, only to be told that this product is not available, but a higher price product is.

I certainly think that the marketing approach taken by PET has been a disgrace to our industry and I would certainly expect that magazines, such as *Personal Computing*, which purport to be knowledgeable about the industry would expose, rather than perpetuate, the fraud which PET has unleashed upon us. After 10 months it is certainly time that Commodore either "Put Up" and deliver the $595 computer, or "Shut Up", along with the magazines who perpetuate their bait and switch game.



William E. Regan, Jr.
President, Technico, Inc.
Columbia, MD

Commodore Replies:

In response to the letter from Mr. Regan, I think I should individually answer several of his points.

One, we have never advertised the PET at any price. We announced a 4K model at $595 for the U.S. market and the slightly modified 8K model at significantly higher prices in Europe, England, Canada and Japan.

Because of the demand for the product and because of production delays, we have only been delivering the product to a very limited number of dealers on strict allocations, and on a prepaid basis to individuals directly from Commodore sales offices, or retail outlets.

Although we did fill some early 4K PET *orders, we are not currently delivering the 4K model and will not until our production matches our demand for the higher profit models. We will only advertise the product when we have significant supply to satisfy the demand and have started stocking a significant number of dealers.*

Two, the leaked price of $495 was an attempt to see if an under $500 price was necessary to create the market.

We did not have a booth at the West Coast Computer Faire but instead showed the PET *as a draw for hiring people to work in our retail stores in the Bay area.*

I am sure that the West Coast Computer Faire people will verify that only Mr. Calculator had a booth and that we were attempting to obtain store personnel and sell the KIM-1, *a product we were delivering.*

Based on the Faire and other marketing inputs we felt that a $600 machine would do very well and increased the features of the prototype and announced the first PET *at the June* CES *at $595 for the 4K and $795 for the 8K version.*

Although we expected a very positive response to the product and had planned a significant buildup of the product, we did not expect the overwhelming response that we received.

In order to better match the demand to our production, we decided to limit dealer participation to those already selected and to require a 100% deposit on individual orders.

We also expected that most of our initial purchasers would be more sophisticated purchasers who would recognize the benefit of the increased memory size and who would be ordering the 8K version, so we scheduled our production for mostly 8K systems.

When we ran into production startup problems with component availability, we were forced to even more significant allocation techniques. One of these was to de-allocate 4K systems and to offer to refund the money for those people who ordered them.

A second was to delay overseas deliveries and then to significantly increase the foreign price of the 8K system.

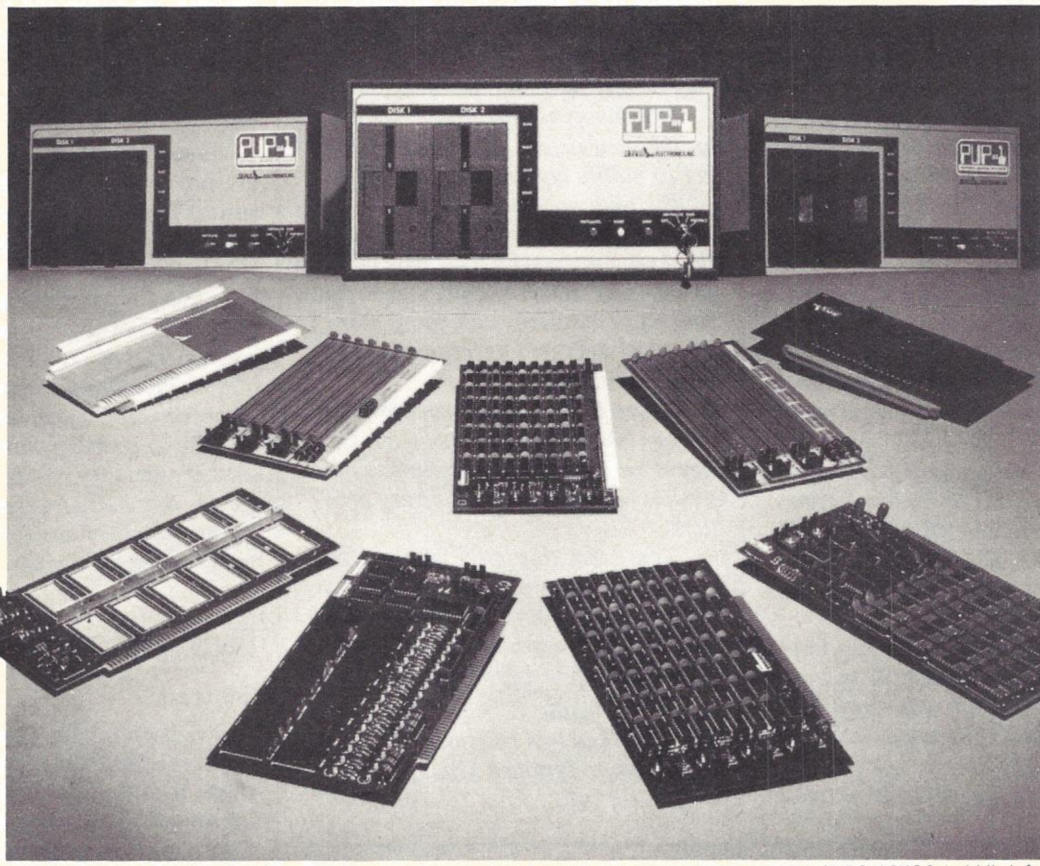
We are currently still having to allocate products to make sure that we serve all markets fairly.

Although we expect to be shipping adequate products to allow dealer deliveries in April, we may be in an allocation situation until our two overseas facilities come on stream in midyear.

So, our problem is not one of bait and switch, our problem is one of not having enough products.

(Continued on following page)

What you should know about the Seals Retail Program.



The performance-packed family of Seals products featuring (lower left to right) BBUC, 4KROM, 8KSC, 16KSC, (middle left to right) 68EXT large and small, 68WWC, 68KSC, 88WWC, 88EXT (top) PUP-1 computer.

At Seals Electronics we believe in specialization. We concentrate on manufacturing a reliable line of microcomputer products so that you, the retailer, can concentrate on sales.

When you are a Seals dealer you can depend on Seals for:

- Dependable, quality hardware with an industry wide reputation for excellent performance.
- A marketing program designed to support you at the point-of-sale.
- Price margins that give you the best possible advantage and recognize that you need profits to operate.
- A staff of marketing and technical specialists who are interested in you and your business.

The people at Seals have been in the microcomputer business for a long time. We know that the computer retailer needs and deserves the very best possible support from a manufacturer. We are constantly on the alert to improve our retail program. Seals works with you to increase your sales and assure you satisfied customers. We don't ask that you be a manufacturer (all our products are available assembled and tested) and we know you don't want us to be a retailer.

The 1977 Computer Store Survey published by Image Resources gave Seals Electronics consistently high ratings in the areas of product image, value to customers (product reliability and documentation), and dealer interface with manufacturers.

We are proud of our record with retailers and are working hard to improve our position in the industry. We would like to work with you.

For current literature on the Seals microcomputer product line and/or more information on our retail program, call or write our Marketing Department, Seals Electronics, 10728 Dutchtown Road, Concord, TN 37922, (615) 966-8771.

Dealer support is more than just words to us.

SEALS ELECTRONICS®
INCORPORATED

CIRCLE 9

The letter raises the question of why should we allocate products so as to make maximum profit.

We are a publicly held corporation whose stockholders have every right to expect that we will attempt to sell each of our products at a maximum profit. It is only because of our commitment to the long term market that we continue to offer the current product mix during the time of limited product availability.

Although the 4K PET *is a very strong computer, the 8K is about 16 times stronger because the system is more balanced. So, a 30% increase in price for that much performance increase is quite fair.*

Although I am not going to discuss our product costs, I invite Mr. Regan's quote to upgrade our 4K system with the same quality level of 4K Static ROMs for $20. I could use the production help.

We do expect that as software develops, more people will buy 8K systems. However, we have and will continue to sell both products at competitive prices.

If Mr. Regan feels that our prices are too high I invite him to offer a product that is competitive at a lower price.

He is raising a serious question. Is it possible for a magazine to attempt to report on exciting new products without competitors, whose market you may be affecting, complaining?

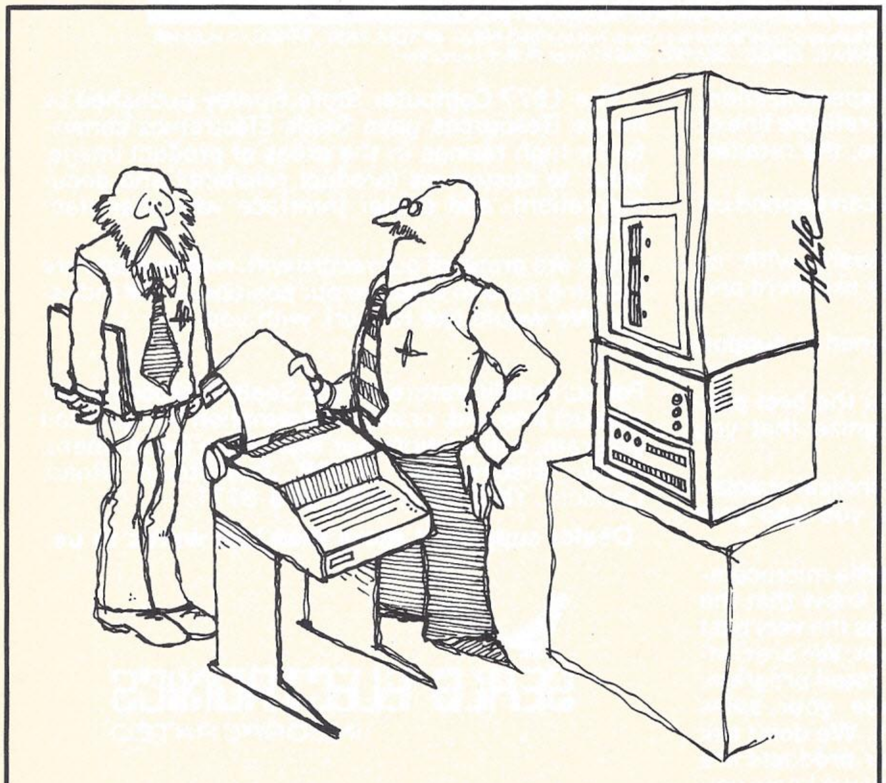
How many products must we have on the shelf before we announce a product to nullify anyone's claim of fraud?

The only people who have any complaint about Commodore's delivery are those customers who have paid and had to wait and our stockholders who know that we could have made more money if we had delivered as fast as people want machines.

We are all working to support the people. We are now under 60 days on direct orders and have begun to supply dealers with immediate delivery products.

I hope that this letter will help to clarify the status of the PET *and allow all of us to get on to the next problem of where are we going to find enough peripherals to satisfy the market.*

Charles I. Peddle
Director, Systems Division
Commodore Business Machines, Inc.
Palo Alto, CA



Editor's Note: In response to the communication between Technico and Commodore, we contacted a number of people we know who have ordered PETs. After varying lengths of time, some have received their purchases. Others are still waiting.

As we are concerned with our readers' interests and with claims of fraud against manufacturers, we felt it necessary to investigate this issue. From our evaluation of the situation, we believe that Commodore is not trying to pull "bait and switch" practices on consumers, but rather they are victims of a problem many manufacturers face – meeting the demand with supply . . . a common problem when a new product catches the fancy of the public to a greater degree than expected. Commodore has been delivering and they are a reputable concern.

Robotic Interest

Editors:

I've been thinking about building a robot for some time but did not want to get involved with welding. After reading your article on constructing and programming a robot (January 1978), I want to find out more about Mr. Newhouse's project.

Could you send me the address of the the fischertechnik people? Do they sell to individuals or through the mail or through retail stores? Do they have a catalog? Do you know any other sources of similar construction material?

I would also appreciate the names of any other useful suppliers of electromechanical parts, etc. Are there any practical books on interfacing, etc. which are of any use?

I have a good theoretical background but not much practical experience.

Alan Filipski
Tempe, AZ

Author's Note: The fischertechnik parts used in the construction of "Bert" were obtained through retail stores. They have a catalog which can be obtained from: fischertechnik, c/o Mr. Kieswetter, International Playthings, Inc., 151 Forest St., Montclair, NJ 07042.

I was pleased with the results obtained by using fischertechnik materials. However, the robot "Bert" is extremely limited in terms of strength, speed and

JADE

FOR ALL CUSTOMERS EXCEPT CALIF. CALL TOLL FREE 800-421-5809

21L02 (450ns) Static Rams 100 @ $1.10 ea.	1702A E—PROM 8 @ $3.75 ea.	6502 Microprocessor 5 @ $11.00 ea.	2708 (450ns) E—PROM 8 @ $11.00 ea.	21L02 (250ns) Static Rams 100 @ $1.36 ea.	Z—80A Microprocessor 5 @ $25.00 ea.	8212 8 Bit I/O Port 25 @ $3.00 ea.	4116 (200ns) 16K Dyn. Ram 16 @ $24.00 ea.
Z-80 Microprocessor 5 @ $20.00 ea.	8224-4 Clk.Gen.&Dvr. 25 @ $8.75ea.	410D (200ns) Static Ram 100 @ $8.75ea.	4096 Dynamic Ram 100 @ $3.50 ea.	4200A (200ns) Static Rams 25 @ $10.00 ea.	74LS367 Hex Buffer 100 @ .70¢ ea.	74LS368 Hex Inverter 100 @ .70¢ ea.	2513 (5v) Character Gen. 5 @ $9.00 ea.

MICROCOMPUTER COMPONENTS

MICROPROCESSOR'S

F8	16.95
Z80	22.00
Z80A	28.00
CDP1802CD	19.95
2650	24.95
AM2901	22.95
6502	11.95
6800	18.95
6802	25.00
8008-1	9.95
8035	22.00
8080A	11.95
8085	27.00
TMS9900TL	75.00

8080A SUPPORT DEVICES

8212	3.50
8214	9.00
8216	3.75
8224	3.50
8224-4	9.95
8226	3.95
8228	7.95
8238	7.50
8251	9.95
8253	21.95
8255	21.95
8257	21.95
8259	21.95
8275	75.00
8279	20.00

FLOPPY DISC CONTROLLER

1771B	55.95
1771B-01	57.95

KEYBOARD CHIPS

AY5—2376	13.75
AY5—3600	13.75

PROM'S

1702A	4.00
2704	15.00
2708	12.00
2716	30.00
2716 Intl	38.00
2758	26.60
D3601	4.50
D3604	13.00
5203AQ	5.00
5204AQ	7.50
6834	17.50
6834-1	14.95
82S23B	4.00
82S129B	4.25
8223B	3.50

6800 SUPPORT

6810P	4.95
68B10P	6.00
6820P	7.50
6821P	7.50
6828P	11.25
6834P	16.95
6850P	9.75
6852P	11.75
6860P	10.00
6862P	14.50
6871P	28.00
6875P	8.75
6880P	2.50

Z80 SUPPORT DEVICES

3881	12.95
3882	12.95

STATIC RAMS

21L02	1.50	1.24	1.18
21L02 (350)	1.60	1.35	1.25
21L02 (250)	1.75	1.60	1.50
410D	10.75	10.00	9.25
1101A	1.00	.90	.80
2101-1	2.95	2.75	2.60
2102	1.25	1.15	1.00
2111-1	3.95	3.50	3.25
2112-1	2.95	2.80	2.69
2114-3	11.00	10.00	9.25
2125L	11.10	9.00	8.30
2147	37.50		
31L01	2.50	2.35	2.00
3106	3.95	3.70	3.25
3107	3.95	3.70	3.25
TMS-4044	9.95	9.00	8.95
4200A	12.95		
TMS-4045	11.00	10.00	9.25
5101	8.30	7.40	7.25
74C89	3.25	3.05	2.85
7489	2.25	2.10	1.90
74S201	4.50	4.00	3.75
P8101	4.20	3.40	2.80
P8155	17.00	14.00	
P8156	21.00	18.00	
8599	1.88	1.75	1.60
9102BPC	1.65	1.45	1.30

CHARACTER GENERATORS

2513	6.75
2513 5v upper	9.75
2513 5v lower	10.95
2516	10.95
MCM6571	10.95
MCM6571A	10.95
MCM6574	13.25
MCM6575	13.25

WAVEFORM GENERATOR

8038	3.50
MC4024	2.25
566	1.50

DYNAMIC RAMS

416D/4116	32.00
1103	1.00
2104	4.00
2107B	4.25
2107B-4	3.95
TMS4050	4.00
TMS4060	4.50
TMS4070-2	32.00
4096	4.00
4116/416D	32.00
MM5270	4.50
MCM6605	5.00

USRT

S2350	10.75

UART'S

AY5-1013A	5.25
AY5-1014A	8.25
TR1602B	5.25
TMS6011	5.95
IM6402	10.80
IM6403	10.80

MISC. OTHER COMPONENTS

N8T20	3.39
N8T26	2.10
N8T95	1.35
N8T96	1.35
N8T97	1.35
N8T98	1.35
81LS95	2.00
81LS97	2.00
1488	1.75
1489	1.75
D3205	4.00
D3207A	4.55
D3208A	14.20
D3211	10.00
B3222	9.75
B3242	10.15
D3245	5.60
C3404	6.75
P3408A	12.00
P4201A	5.20
MM5320	7.50
MM5369	1.90
TMS5501	24.95
DM8130	2.90
DM8131	2.75
DM8833	2.50
DM8835	2.50
DM8837	1.75
MK50240	20.00
MK50250	15.00

JADE 8080A KIT

$100.00 KIT

BARE BOARD $30.00

E—PROM BOARDS

MR 8 (8K uses 2708) with 1K RAM	KIT	$99.50
MR 16T (16K uses 2716) with 1K RAM	KIT	$99.50
MM 16 (16K uses 2708)		$99.00
RAM/N/ROM (16K uses any E PROM)	KIT	$117.00
JG 8/16 (uses 2708 or 2716)	KIT	$59.95
BARE BOARD		$30.00

TU-1

Convert T.V. set to Video Monitor.

KIT $8.95

STATIC RAM BOARDS

8K

250ns ASSEMBLED & TESTED	$189.95
450ns ASSEMBLED & TESTED	$149.75
250ns KIT	$169.95
450ns KIT	$125.00
BARE BOARD	25.00
6800 ADAPTER to S-100 System KIT	$12.95

16K

250ns ASSEMBLED & TESTED	$435.00
450ns ASSEMBLED & TESTED	$380.00
450ns KIT	$335.00

32K

250ns ASSEMBLED & TESTED	$850.00
450ns ASSEMBLED & TESTED	$775.00
450ns KIT	$675.00

DYNAMIC RAM BOARDS

On board Refresh power is provided with no wait states or cycle stealing required. +8VDC 400MA DC, +18VDC 400MA DC and −18VDC 30MA DC.

EXPANDABLE 32K

8K (375ns)	KIT	$151.00
16K (375ns)	KIT	$259.00
24K (375ns)	KIT	$367.00
32K (375ns)	KIT	$425.00

EXPANDABLE 64K

16K (375ns)	KIT	$281.00
32K (375ns)	KIT	$519.00
48K (375ns)	KIT	$757.00
64K (375ns)	KIT	$995.00

MOTHER BOARD's — S-100 Style

13 slot — w/front panel slot

BARE BOARD $35.00

KIT $95.00

22 slot $149.95

ASSEMBLED & TESTED

THE PROM SETTER

WRITE & READ

EPROM

1702A — 2708 — 2716

5204 — 6834

- Plugs directly into your ALTAIR IMSAI Computer
- Includes Main Module Board and External EPROM Socket Unit
- The EPROM Socket Unit is connected to the Computer through a 25 pin connector
- Programming is accomplished by the Computer
- Just read in the Program to be Written on the EPROM into your Processor and let the Computer do the rest
- Use Socket Unit to Read EPROM's Contents into your Computer
- Software included
- No external power supplies Your computer does it all.
- Doubles as an Eight Bit Parallel I/O
- Manual included

KIT $210.00

ASSEMBLED $375.00

KIM—1

ASSEMBLED & TESTED $245.00

MEMORY PLUS

for KIM—1

8K RAM (21L02)

8K EPROM

ASSEMBLED & TESTED

$245.00

JADE Z80 KIT

—with PROVISIONS for ONBOARD 2708 and POWER ON JUMP

$135.00 EA. (2MHZ)

$149.95 EA. (4MHZ)

BARE BOARD **$35.00**

JADE Z80 "UPGRADE" KIT

Change your JADE 2MHz Z80 to a 4MHz version with this simple kit:

only $17.95 with trade

$49.95 purchase.

To trade, you must give us your 2MHz Z80 chip and 8224 clock driver.

The "UPGRADE KIT" includes:

Z80A chip,	1.8K resistor
8224-4 clock driver	20 pf. capacitor
36MHz crystal	

COMPU/TIME CT 100

S100 BUS COMPATIBLE

TIME & CALENDAR

Microprocessors need the power that a real-time clock can offer. Date and time becomes instantly available. **COMPU/TIME** does not have to be initialized every time the system is powered up. It possesses a crystal-controlled time base to obtain superior accuracy and has two setable coincidence counters. Time, date, and counters are set via software.

COMPUTATIONAL FUNCTION

Microprocessors need to be complimented by hardware arithmetics to free up memory pages dedicated to floating point routines and mathmatical software. **COMPU/TIME** provides a 40 function calculator array so that algebraic, trigonometric, basic arthmetic problems can be solved without the need of developing sophisticated software.

Buy It Your Way

COMPU/TIME	CT100	$199	KIT	$245	Assembled
COMPU only	C101	$149	KIT	$189	Assembled
TIME only	T101	$165	KIT	$205	Assembled
COMPU/TIME	PC Board only	$80			

JADE VIDEO INTERFACE KIT

FEATURES **$99.95**

S-100 Bus Compatible
32 or 64 Characters per line
16 lines
Graphics (128 x 48 matrix)
Parallel & Compositive video
On board low-power memory

Powerful software included for cursor, home, EOL, Scroll Graphics/Character, etc.
Upper case, lower case & Greek
Black-on-white & white-on-black

full ASCII PROFESSIONAL KEYBOARDS

Full 128 Character ASCII!
Tri-Mode MOS Encoding!
MOS DTL TTL Compatable Output!
Two-key Rollover!
Level and Pulse Strobe!
Shift and Alpha Lock!
Selectable Parity!
Positive or Negative Logic!

MODEL 756 (56 keys)

PRICING INFORMATION

Model 756 (assembled)	$75.95
Model 756K (kit)	64.95
Model 702 Enclosure	29.95
Model 710 Numeric Pad	9.95
Model 756MF Mtg. Frame	8.95

CONNECTORS

DB — 25P	$3.00
DB — 25S	$4.00
COVER	$1.50
44 Pin — PC & EYE	$1.95
44 Pin — WW	$2.50
86 Pin — (6800) PC	$5.00
86 Pin — (COSMAC ELF) PC	$5.00
100 Pin — (Altair) PC	$4.50
100 Pin — (Imsai) PC	$3.75
100 Pin — (Imsai) WW	$4.25

WE HONOR master charge THE INTERBANK CARD

JADE REAL TIME CLOCK FOR S-100 BUS

1 MHZ Crystal Oscillator
Two independent interrupts
One interrupt uses 16 bit counter in 10 USEC steps
Other interrupt is in decade steps from 100 USEC to 10 sec.
Both software programmable
Board can be selected by 128 device code pairs.
Complete documentation includes software to display time of day.
Double sided solder mask
Silk screen parts layout

JG—RT ASSEMBLED & TESTED	$179.95
JG—RT KIT	$124.95
BARE BOARD with Manual	$30.00

TARBELL CASSETTE INTERFACE

- Plugs directly into your IMSAI or ALTAIR*
- Fastest transfer rate: 187 (standard) to 540 bytes/second
- Extremely Reliable—Phase encoded (self-clocking)
- 4 Extra Status Lines, 4 Extra Control Lines
- 37-page manual included
- Device Code Selectable by DIP-switch
- Capable of Generating Kansas City tapes also
- No modification required on audio cassette recorder

JADE KIT* $99.95 ASSEMBLED $175.00

*(6 month warranty from JADE) MANUAL $4.00

JADE PARALLEL/SERIAL INTERFACE KIT

S-100 **$124.95 KIT**

2 Serial Interfaces with RS232 interfaces or 1 Kansas City cassette interface.
Serial interfaces are crystal controlled.
Selectable baud rates.
Cassette works up to 1200 baud.
1 parallel port.

DATA COMMUNICATIONS ADAPTER

80-103A Serial I/O and FSK modem for professional and hobby communications.

- Completely compatible with your IMSAI, ALTAIR* SOL** or other S-100 microcomputers. Trademarks of *MITS, **Processor Technology
- Designed for use on the dial telephone or TWX networks, or 2-wire dedicated lines, meets all FCC regulations when used with a CBT coupler.
- All digital modulation and demodulation with on board cyrstal clock and precision filter mean that NO ADJUSTMENTS ARE REQUIRED
  - Bell 103 standard frequencies
  - Automated dial (pulsed) and answer
  - Originate and answer mode
  - 110 or 300 BPS speed select
  - Complete self test capability
  - Character length, stop bit, and parity
  - 90 day warranty and full documentation

PRICES:	
BARE BOARD and Manual	$49.95
Assembled (48 hr. burn in)	$279.95
JG—DCA KIT	$159.95

NUMBER CRUNCHER

The CT200 is a number-oriented microprocessor intended for use in those applications that require fast versatile mathematical solutions.

THIS IS NOT A CALCULATOR CHIP. THERE ARE NO KEY DELAYS.

The CT200 has a unique architecture that is designed to be a TASK processing system within a system. This unique architecture will allow the CT200 to work and run with ANY S100 BUS microprocessor system. It is completely compatible with Z80, 4MHZ version also, 8080, 6800, 6502 microprocessor. A micro incoded instruction set allows programming in a calculator like language. The instruction set includes a full set of test and branch instructions. All decoding of S100 bus signals (for select or control functions) is performed with strobed latches to eliminate the possibility of glitches.

PRICE: $249.00

Includes · Manual, ASSEMBLED & TESTED.

JADE Computer Products

BANKAMERICARD VISA

5351 West 144th Street
LAWNDALE, CALIFORNIA 90260
(213) 679-3313

RETAIL STORE HOURS: Monday · Friday 9—7, Saturday 9—5

Discounts available at OEM quantities ADD $1.50 under 10 lbs. for shipping. California residents add 6% sales tax.

NEW CATALOG NOW AVAILABLE

See us at NCC Booths 94, 95, 96, 97

CIRCLE 10

weight-carrying ability. The structure is relatively fragile.

Also, the motors (8) in the robot each cost around $40 including a large kit of other parts. You just cannot buy the motors economically.

However – do not despair! I have been assembling the parts necessary to build a strong, practical and reasonably inexpensive robot which will utilize the same basic structure as Bert. It will use powerful gear drive motors which only cost $12 each. The structure will be capable of transporting 500 lbs., will be approximately 3' x 5' with an on-board computer, video recorder and wireless remote control.

With complete construction drawings of each metal part, you will be able to take the plans to a machine shop and execute them in steel.

I also plan to write a listing of the BASIC language program (a compiled program for high speed).

– Sam Newhouse

Language Jam

Dear Editor:

Regarding your March story on Computer Languages, it seems to me that Jef Raskin is talking like a Latin teacher to his students. You've got to "Homni Dominus" your "Marcus Timetis" if you want to get anywhere, kids! Baloney! With all these languages cluttering up the computer rooms, it's beginning to sound like a traffic jam around the Tower of Babel.

Now, I happen to know that the Data Group has a machine that you can talk to in uncomplicated simple English. Furthermore, you can start using the computer the minute you get it uncrated. I can tell you that the microprocessing industry will continuously drag its feet until the day it smartens up.

Turn out computers that listen to you in your own language and let the machine's compiler or assembler translate it any way it wants. If you had to learn special languages to operate a stereo system, we'd all still be back in the days of the Victrola.

Give the people something they can use the first day they get it home and the personal computer industry will leap forward. Why should we have to translate our language into the language of the computer. Why can't the smart-alecky computer do its own translation?

Goethe would still be an unknown poet if we all had to learn German before we could read his poems. Mit grossern verdanken!

George Hildreth
Chicago, IL

Try It Yourself

Dear Editor,

I really enjoyed the double-barrelled cartoon on p. 71 of the January 1978 issue. Double-barrelled? Well, have you thought of converting 47806_{10} to hexadecimal? Do it!

Joachim F.L. Sommerfeld
Coon Rapids, MN

Taking Stock

Editors:

Regarding your article on the stock market (February 1978), you might add this point. North Atlantic Industries' (manufacturers of magnetic tape storage systems and electronic equipment) recent 50% annual growth rate, created by the Quantex Division's sales of cartridge storage systems, would make this firm fit one of your hypothetical scenarios.

If you had attended the meeting of the Boston Stockbrokers Club, you would have been able to gauge the stockbrokers' enthusiasm and judge for yourself.

Incidentally, the stock market closes at 4 p.m., not at 3, as noted in your article.

Stanley Froud
Plainville, NY

More Floppy Facts

Dear Editors:

I enjoyed "Facts on Floppies" in your March issue. Good information!

I especially appreciated the footnote regarding the BASIC compiler. That's what I needed.

Can you give me an address for Electronic Product Associates?

Reginald Creighton
Washington DC

Electronic Product Associates, Inc. is located at 1157 Vega St., San Diego, CA 92110.

Case in point

Dear Editors:

In your "Linear Programming" article (January 1978), you specify at line #150, 235 and 315 an input of B1, Y1; however, in your output, there is no indication of the B1 variable's values. Therefore, I am wondering whether or not the output is the product of the listed program or possibly a revision.

Each way I enter the data I get either an "unbounded" error or a reference to a higher constraint than the data calls for. I may be missing some item in the program.

M.E. Motes
Chillicothe, OH

Author's Note: The answer to the first problem mentioned in Mr. Motes' letter can be found in the remarks at the beginning of the program listing.

Mr. Motes also pointed out a problem that occurs when there is no = constraint. When this is the case, A2 will be zero and the program will indicate an unfeasible solution. This is because line 525 should read: 525 IF X(Y1)<=0 THEN 672.

– Paul Whittington

February Review

Dear Editors,

I read your February issue with some alarm. Its articles were mostly short and trivial.

However, there were some articles of real quality. The interview was truly fascinating. It dealt effectively with an important issue, the purpose of the computer, a valuable service. The robot article was also interesting since I own a very large set of Fischer-tech.

However, the overall quality of the articles was poor. The games were ridiculously simple. The article on RAM boards was useful but incomplete. The final article, "Computers in Developing Nations", was stimulating but did not bear directly on personal computers. Reprinting is a dangerous practice, especially if you have only 13 articles.

Columns are important. You need editorials and involved comment. The magazine as it stands simply does not say anything. I read most of the published computer magazines and yours needs the most work.

New! **SELECTERM** SYSTEM 9710

The SELECTRIC II* Printer you can TRUST

BECAUSE . . . It's brand new, and fully assembled and tested.

BECAUSE . . . After extensive engineering design and testing by Micro Computer Devices, IBM Corporation has approved the SELECTERM for use with your microcomputer, and provides you with their factory warranty and yearly service agreement for the typewriter. In addition, the electronics conversion portion is fully warranteed by Micro Computer Devices.

BECAUSE . . . You can connect the SELECTERM to your computer within minutes of taking it out of the carton.

IT'S THAT EASY! AND THAT RELIABLE!

FEATURES
- Complete ASCII character set in supplied element.
- Full upper, lower case alphanumeric characters.
- Tab Command, Index (vertical tab), Backspace, Bell—all under computer control.
- Parallel Interface, standard.

ALL ELECTRONICS INCLUDED
- Power supply, electronics and cable sets included to permit immediate connection to the parallel port of any computer, at standard TTL level.

SOFTWARE
- All necessary conversion software in PROM to handle ASCII input, directly.

PRINTER or TYPEWRITER
- May be used as a standard typewriter when not in use with your computer.

OPTIONS
- Dual Pitch, $125
- Correction Feature, $125
- Tractor Feed Platen, $250
- Noise Reduction Feature, $50

AVAILABLE SOON
- RS-232 Interface

PRICE and DELIVERY
- Assembled and tested, $1750
- Available ONLY from authorized dealers.
- Delivery 1 to 2 weeks from receipt of order.
- OEM delivery in quantity within 30 days.

micro computer devices inc.
960 E. Orangethorpe, Bldg. F
Anaheim, California 92801
Telephone (714) 992-2270

"Innovators to the Microcomputer Industry"

*Registered trademark of IBM Corporation

CIRCLE 11

How about reviewing companies' performance? It would take up space and would be a real service. People, especially beginners, need to know that the companies they are sending money to are going to send something back.

James A. Glazier
Cambridge, MA

Correction

Dear Editors:

Although I am quite flattered at the attention that *Personal Computing* has given the PET and the accuracy of their interview about the PET and Commodore's plans for this market (Sept/Oct 1977), there was a total misunderstanding when we were discussing the developments of the Motorola 6800 microprocessor and the MOS Technology 650X.

As a proud member of both these development teams I would like to put the record straight about who developed these products.

Although no one now develops a product of the complexity of a microprocessor, the major credit for the Motorola 6800 must go to Tom Bennett from Motorola who was team leader and chief architect and Buck Buchanan (currently with NCR) who was project engineer. Other key contributors were Earle Carlow, Mike Wilde, Rod Orgil and Tony Kauosas, as well as a support team of about 30 people, all of whom made a more significant contribution than I to that product.

Major credit for the 650X should go to Will Mathis, who did most of the architecture, and Rod Orgil (also currently at NCR) who was the project engineer, and Bill Mench, who did much of the circuit design and detailed logic, Ray Hirt, Terry Holt (currently with AMI), Harry Bowcon (currently with Motorola), Mike Jones (currently with NCR) and many others who again made the 650X a successful product.

I hope by listing these gentlemen and asking you to publish this we can clear me from claiming that I could have done any of their work and can concentrate on the PET and its contribution to making computing personal.

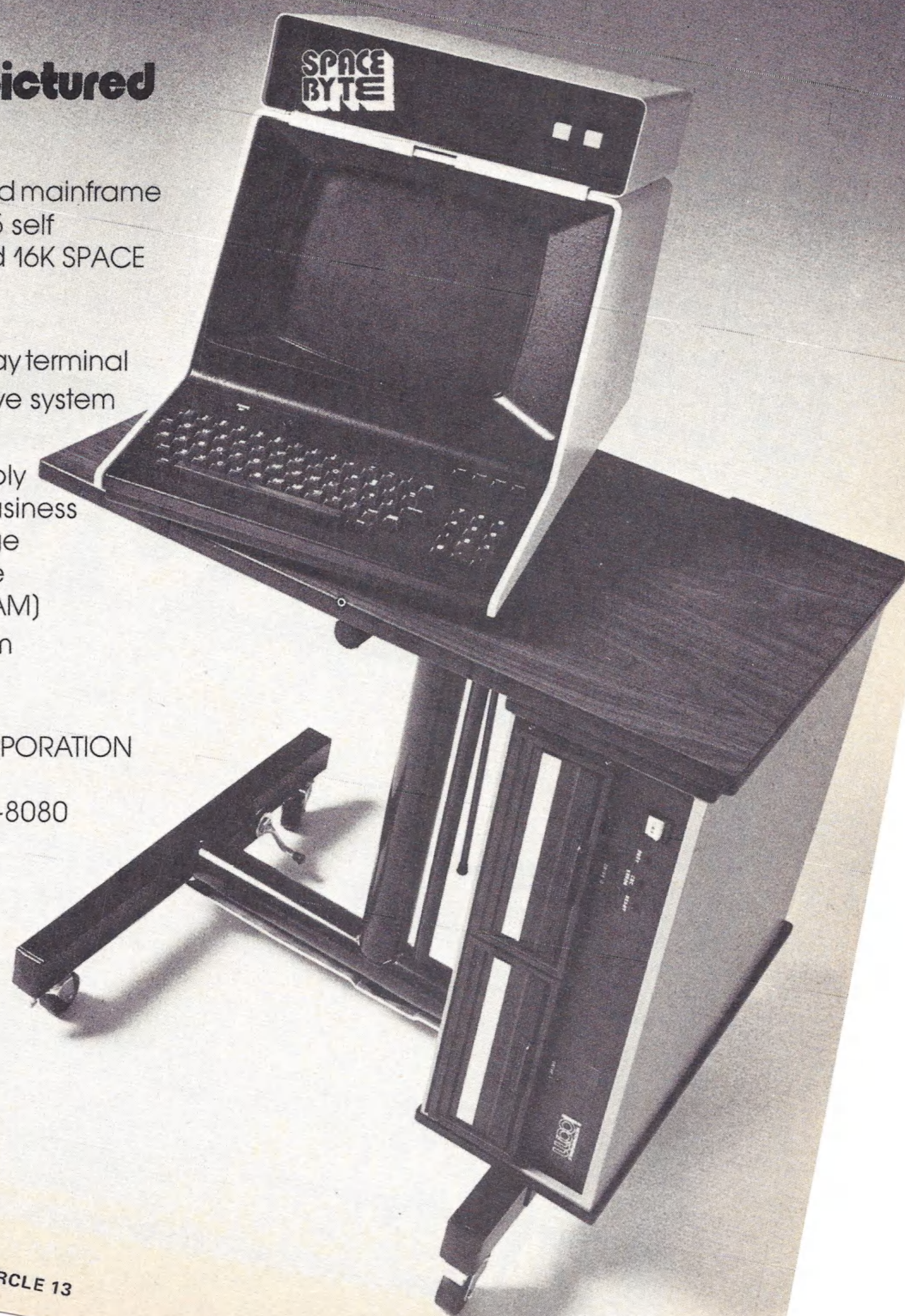
Charles I. Peddle
Director, Systems Div.
Commodore Business Machines, Inc.

Comments, questions and criticism are welcomed

All letters should be sent to: *Personal Computing*, 1050 Commonwealth Ave., Boston, MA 02215. All material is subject to editing for length, libel and standards of good taste. Signatures may be withheld upon request.

Introducing the SPACE BYTE MODULAR BUSINESS COMPUTER

Complete with assembly language application software



Complete as pictured $5,900.

- SB85-16 terminal mounted mainframe with the SPACE BYTE 8085 self contained computer and 16K SPACE BYTE fully static RAM (48K RAM capacity)
- HAZELTINE 1500 video display terminal
- Dual iCOM flexible disk drive system
- Floor stand
- SPACE BYTE "BIZPAK" assembly language disk interactive business application software package optimized by the speed of the 8085 CPU (requires only 16K RAM)
- iCOM FDOS III operating system

THE SPACE BYTE COMPUTER CORPORATION
1720 Pontius Ave., Suite 201
Los Angeles, CA 90025 (213) 468-8080

CIRCLE 13

Booking It

I'm trying to locate books on different languages (PASCAL, LISP, ALGOL, etc.) Any suggestions?

Neil Freeman
14 Margpret Drive
West Lawn, PA 19609

Say Ahhhhh . . .

I'm interested in information that would be helpful in purchasing and operating an in-house computer system with the capability of handling all facets of a busy dental practice, i.e., medical records, accounts receivable, generating the monthly billing, insurance forms and supply inventory.

Robert Barclay, DDS
Pemasco Dental Clinic
P.O. Box 15
Pemasco, NM 87553

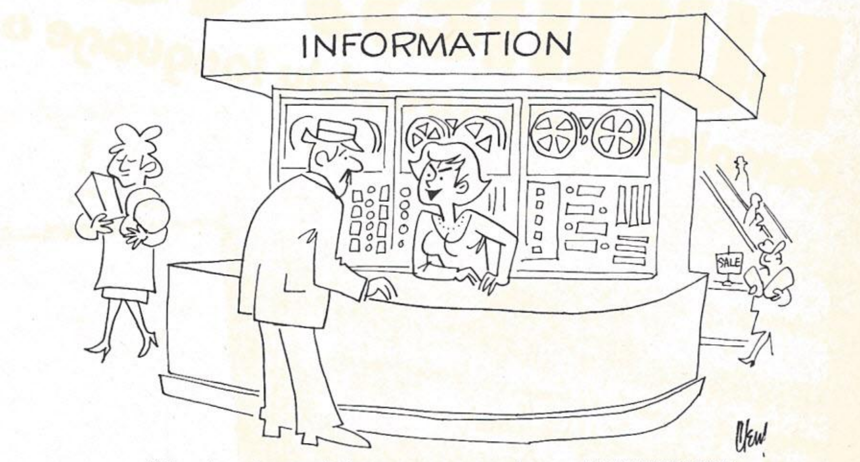
Import/Export

I'm interested in understanding the practical uses of personal computers for home and business.

I'm going into the exporting/importing of retail goods and will need a small computer to handle customer identifications, inventory, prices, orders, etc.

I'd appreciate all help anyone can offer.

William Randle
227 Ashbury St.
San Francisco, CA 94117



"Go ahead . . . . I dare you . . . . ask me ANYTHING!"

With a little imagination and a bit of hardware, you can teach your computer to read handprinted characters.

PATTERN RECOGNITION

BY ALAN FILIPSKI

Human beings, even young children, are extremely efficient classifiers of visual objects. They can recognize a picture of a person, for example, whether it's a stick-figure drawing, a photograph, or a water-color painting. Furthermore, the pictured person may be wearing any sort of clothing, be upside-down, or have his head in a bag. None of these variations offers the least difficulty in recognizing the picture as that of a person.

It's extremely difficult, perhaps impossible, to describe in precise detail (the kind of detail computers like) exactly what characterizes a picture of a human being. For any well-defined characteristic we select, someone can probably create a picture of a person which does not possess that characteristic. Humans, nevertheless, are able to make such classifications consistently and within fractions of a second.

What method does the human pattern classifier use that is so hard to formalize? Even if we don't understand how a person does it, can we make machines (our computers?) perform as well at this sort of task? The answer to these questions lies in a multidisciplinary area known as pattern recognition.

Even though pattern recognition draws on psychology, computer science, statistics and philosophy, with with a little imagination and a bit of hardware you can teach your home computer pattern recognition.

Pattern recognition, of course, is not restricted to classifying visual images. It applies to correct classification of spoken words, diagnoses of diseases on the basis of given symptoms, or deciding whether a given chess position is winning. In all these cases, it's unlikely that one would have encountered the *exact* same sound pattern, set of symptoms, or arrangement of chess pieces before.

To construct a machine able to draw these kinds of distinctions, we must first ask how a person acquires this ability.

One of two possible approaches involves being told exactly how to make the decision. We might tell the person that "brillig" means "four o'clock in the afternoon", or "if you cough and ache, then you have the grippe", or "a bishop and a king cannot force mate against a lone king".

The second, more sophisticated approach allows the individual to learn from experience. We could put the person in a situation where people around him use "brillig" in everyday conversation, or we could show him coughing people complaining of aches and "the grippe", or we could let him play chess games. This second approach requires higher mental faculties than the first, since it requires the person to make inductive inferences.

But neither approach is completely trivial to computerize. "Telling a computer" is not accomplished as easily as "telling a person". Nevertheless, if we choose a suitable problem, we can implement both of these approaches.

Let us consider a simple, concrete example, namely, that of distinguishing between a handprinted 1 and a handprinted 0. If our representation allows us to compute the height and width of the character in

question, we can give the computer the rule "decide 0 if the ratio width/height exceeds some threshold; otherwise decide 1." This rather simple rule will confuse skinny 0s with fat 1s.

If, on the other hand, we wanted to have the computer learn the difference between 0s and 1s without giving it any explicit rules, we could write a program which would read in a set of training samples, that is, properly labelled 0s and 1s. We'd then give the machine the representation of an unidentified symbol and ask for a classification.

One way to accomplish classification is to apply a similarity measure between the new symbol and each set of known symbols and decide on the basis of greatest similarity.

Now let's look at the details involved in constructing actual character recognition systems, both with and without learning.

Extracting the raw data

The first practical consideration is how to input and represent raw data. There are essentially two choices here – a television-type raster scanner or an X-Y digitizer.

In the raster input mode, the image (in this case, a hand-lettered character against a white field) is considered to be superimposed on a grid, and the grid is scanned row by row, each row from left to right. When a cell contains a black mark, a 1 is transmitted; otherwise a 0 is transmitted. Therefore, we get a square square array of bits. We can implement input by interfacing a television camera with a microcomputer.

This setup's principal advantage is that we can draw characters off-line. A disadvantage is that the representation is too general. Usually generality is good, but in this case there's a simpler, more economical representation of the simple handprinted characters. A structure capable of representing complex, shaded objects just complicates our work.

Another disadvantage of raster-scan input is loss of dynamic information. For example, suppose we wish to determine whether the character in Figure 1 is a 4 or a 9. A raster scan of the completed image would leave some ambiguity; but if we knew that the writer lifted his pen after drawing about half of the figure, we would be more certain we had a 4.

The other major alternative, the X-Y digitizer, eliminates the disadvantages of the raster scan. This device converts the hand-printed character in real time to a stream of X-Y coordinates. Of the many such units available, the most economical and practical for small-scale use is the Summagraphics bit pad, so we'll use this unit as an example.

The bit pad consists of an 11-inch-square drawing area and a hand-held stylus. As the stylus moves over the surface of the pad, coordinate pairs travel through an interface to your microcomputer at some selected repetition rate (1 to 64 pairs/sec). Resolution (.1 mm) is sufficiently fine to encode small characters, but larger characters give smoother representation.

Sampling rates of 64/sec produce good results with characters several inches tall. Gaps between strokes may be easily detected and inserted into the stream as negative coordinate pairs. (Actual coordinates are always positive.) Thus, we record the fact that the pen is lifted in drawing a 4.

Another advantage of X-Y representation is its variable length – shorter characters require less storage.

Finally, X-Y stream representation may be easily converted into bit array representation, but the reverse is not true. Because of these advantages, we'll assume we have the bit-pad-type representation available.

Feature Extraction

Raw data usually needs some preprocessing before we can apply classification methods. Preprocessing consists of converting raw data into a set of numerical measurements or into a list of facts which are true about the image. The former are called "numerical features" and the latter, "binary features".

In general, extracting these features results in a loss of information – we cannot exactly reconstruct the original image from the features. But this loss is all right as long as we retain as much information

Figure 1 Is this a 4 or a 9?

relevant to the object's classification as possible. Thus, feature extraction compresses useful information and discards irrelevant information.

Here is a sample set of binary features which may be extracted from the X-Y stream representation of a handprinted character:

GAP – Stream has a gap in it. (Stylus is lifted from pad before character is complete.)
ENDHZ – End of last stroke is roughly horizontal.
CLOSED – First point of stream is approximately equal to last point.
STARTR – First few points of the stream move to the right.
LOOP1 – Initial point of stream is approximately equal to some point in mid-stream.
LOOP2 – Last point of stream is approximately equal to some point in midstream.
UNDER – First point has approximately the same X-value as last point.

These features are fairly simple to extract from the raw data, given that the programmer supplies reasonable meanings to such words as "approximately" and "near". Two points might be defined as "approximately equal" if the distance between them is less than 10 percent of the character's height, for example. (The distance between two points could be the familiar Euclidean distance, d=SQRT $((X_0\text{-}X_1)$** 2+$(Y_0\text{-}Y_1)$**2), where (X_0,Y_0) and (X_1,Y_1) are the two points.)

We can use numerical features as an alternative to binary features. In this case, raw data is transformed into feature vectors $(f_1,f_2, \ldots f_n)$, where each f_i represents a numerical value computed in some way from the raw X-Y stream. Some possible features are:

WIDTH – Maximum X-value minus the minimum X-value in the stream.
DIST – Distance between first and last points of the stream.
LEN – Total length of the stream.

Creation of such features is limited only by the programmer's imagination. Figure 2 shows feature vectors corresponding to some handprinted 0s and 1s using these three features.

"Probes" are a frequently used method of extracting numerical features from handprinted characters. A probe is an imaginary directed line segment emanating from a given point on the paper. The probe feature's value is the distance the probe travels before intersecting the character.

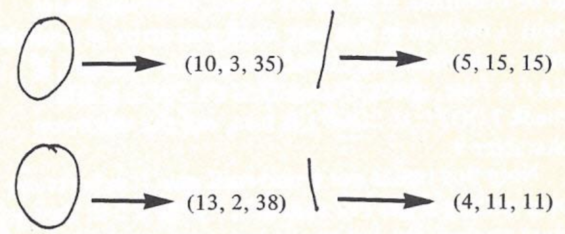


Figure 2 Feature Extraction

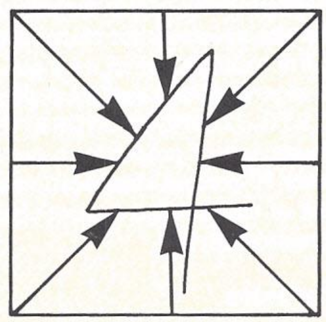


Figure 3 Probe Lines

Figure 3 shows a character with a commonly used set of probes. The feature vector corresponding to this image and this set of probes is $(p_1, p_2, \ldots p_8)$, where each p_i indicates the length of one of the probes. The probe technique may be used with either bit array or coordinate stream representation methods.

Now we can see how pattern recognition may be performed using either binary or numerical features. First, we consider the simplest case, namely, pattern recognition without learning.

Decision Tree

Consider the problem of writing a special-purpose program utilizing some of the above features to determine the identity of an input handprinted character. To keep the size of the example down, let's assume input is restricted to the 10 decimal digits in hand-lettered form and that we use the set of binary features given above.

We can now describe the classification program by the decision tree given in Figure 4. This tree is essentially a flowchart. Starting at the upper left corner, ask whether the given binary feature is true of the image

to be classified. If so, move down; otherwise move right. Continue in this way until you arrive at a single digit – the resulting classification. For example, if GAP is false, check CLOSED; if CLOSED is false, check LOOP1; if LOOP1 is true, call the unknown character 9.

Note this tree is not a universal one; it reflects my own writing style. For example, I write a 5 in two separate strokes, the top stroke last, whereas some people use one continuous stroke. Furthermore, a shorter, bushier tree would make the procedure more efficient, because it would need fewer decisions on the average.

An important property of this technique is that only those features necessary for classification need to be extracted. For example, when we input the 9, we don't need to determine whether it has properties ENDHZ, STARTR, LOOP2, or UNDER, thus saving a considerable amount of processing time. It's not difficult to get over 90 percent classification accuracy with a good set of features and a set of characters drawn in a consistent style.

Programs that learn

The above program uses a predefined decision algorithm that always produces the same results for the same input. If the program is consistently wrong, we can correct it only by reworking parts of the code or adjusting some numbers used in the program.

Now consider the more challenging problem of creating a program that learns from experience. Initially, this program will know nothing about the characteristics of the various digits, nor how many different types of digits there are. We only give the program the ability to transform a raw data stream into vectors of numerical features.

After writing the program, we enter the "training" phase, during which we enter many sample characters, telling the computer the correct classification for each. After sufficient training, we require the program to provide us with correct classifications of new inputs similar to those given during training.

Let's take a simple example. Suppose we set up the program to extract the three numerical features WIDTH, DIST, and LEN described earlier. Now suppose we give the program a set of training samples, some of which we identify as 0s, and some as 1s. One simple way to accomplish the learning is simply to store all the feature vectors of the training set along with the correct classification. Then, when we give the

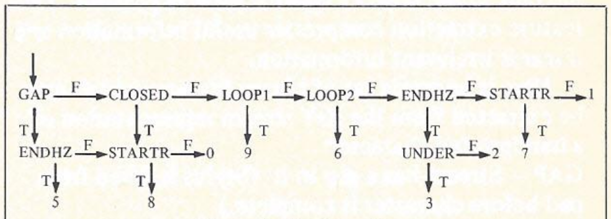


Figure 4 Decision Tree

program an unknown sample to classify, the program can compute the feature vector, compare it to each feature vector in the training set, and classify the unknown character the same as the closest feature vector from the training set.

Ordinary n-dimensional Euclidean distance can provide a measure of distance between vectors. For example, for n features, the distance between ($f_1, f_2, \ldots f_n$) and ($g_1, g_2, \ldots g_n$) is SQRT $((f_1 - g_1)^2 + (f_2 - g_2)^2 \ldots (f_n - g_n)^2)$. This distance measure works very well as long as features are numbers of the same rough order of magnitude, so one number doesn't dominate the others when distance is computed. (We can use a scale factor on the feature if necessary.)

Suppose, for example, our training set consists of the four characters given in Figure 2, and our unknown input has the feature vector (4, 17, 17). Our scheme classifies the unknown as a 1 because the closest vector from the training set corresponds to a known 1. Of course, in a real situation, we would have several classes, more features, and many more training samples.

This classification scheme – the nearest neighbor rule – may be generalized by taking the three nearest neighbors and classifying the unknown based upon the majority of the classifications of its neighbors. This procedure helps correct occasional bad samples in the training data.

But while the nearest neighbor rule performs fairly well, it has some definite disadvantages – for example, it requires storing the entire set of training vectors. Storage can obviously be a problem, in terms of both time and space, with large training sets containing many features.

An alternative is the "prototype" classification scheme – instead of storing all training vectors, we just store one prototype for each class, representing a typical member of that class.

A typical prototype vector would be a vector whose components are the averages of corresponding components of all vectors in the given class. Obviously we

lose some information in this averaging process, with resulting degradation in the accuracy of classification; but savings in time and space can be considerable.

In the example just given, the two prototype vectors are (11.5, 2.5, 36.5) for class 0 and (4.5, 13, 13) for class 1. The unknown (4, 17, 17) is still classified as 1 because it's closer to the first prototype.

Typically, a general learning program doesn't perform as well as a good fixed algorithm on a familiar domain such as handprinted characters. But it's interesting to watch the learning program improve its performance as the training sample size increases.

Also, the learning program is more general, since the same program works on digits, Greek letters or ancient alchemical symbols.

A slight modification allows the learning program to accept "criticism". If it classifies an image incorrectly, the user (teacher) enters the correct classification and requests the program to update its training set. It won't make the same mistake twice. Only the power of the feature set limits ultimate performance.

Further Work

The techniques discussed here provide an introduction to pattern recognition without requiring much mathematical background. Books on pattern recognition contain many other techniques for classification and approaches to answering questions we've avoided raising.

For example, how do we learn that certain features are more important than others for certain classification problems and how do we use that information? How can assumptions about probabilities be used to define classifiers? Can the program learn classes on its own given only unlabelled training samples? (Surprisingly, the answer to the last question is yes and the technique is known as clustering.)

If your taste leans toward experimentation, you can do a lot with the bit pad and a basic knowledge of pattern recognition. It's a challenge to define a set of features which will allow a computer to classify line images as accurately as you or I.

Can you design a system to distinguish your signature from forgeries? What about evaluating expressions written on the bit pad in ordinary algebraic form, with fractions, exponents, radicals, and so forth? How about converting a handwritten flowchart into a program? Advanced pattern recognition works toward all these applications.

RANDOM ACCESS

Old Soldiers Never Die....

In an eleventh hour effort, UCLA's most venerable computer was rescued from the scrap pile and preserved in the Smithsonian Institution.

The UCLA Bush-type Differential Analyzer, installed at UCLA at 1947, is the last working model of its kind. Often described as "an oversized erector set", it was named for and conceived by Professor Vannevar Bush of M.I.T. in 1931. During the late 40s and early 50s the 12-ton, 32-foot by 9-foot computer introduced much of Southern California industry to automatic computing, instructed the first student generation in the new art, and even starred in two Hollywood movies.

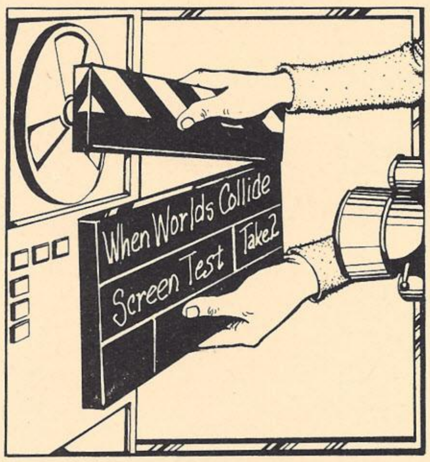
In 1945, when the newly-founded UCLA College of Engineering started shopping for a mechanical analog computer, General Electric offered to construct the differential analyzer for delivery in two years at a cost of $125,000. On July 23, 1947, the new machine was ready to face its first numerical problems, but in contrast to today's computer speeds, it often took hours or days to run off a set of solutions.

Though slow, the differential analyzer was soon fully booked for teaching, research and cooperative contract work with industry. Firms like Douglas Aircraft, Rockwell, Northrop and Hughes got an early taste of automatic computing through the UCLA machine, as did local water and power utilities.

Hollywood also discovered the analyzer. In the era before *Star Wars* and *Close Encounters*, it was the latest in way-out gadgetry and performed skillfully in such science fiction epics of the mid-Fifties as *When Worlds Collide* and *Earth vs. the Flying Saucers.*

But glamour was short lived. With the rapid development of faster and more sophisticated computers, the differential analyzer fell into disuse. From 1960 on, the UCLA engineering school used it mainly as a display piece, although it clanked away occasionally for student and public demonstrations.

Early in 1977, the engineering school decided it needed the space occupied by the ancient computer for other purposes. The chairman of the UCLA Computer Science Department, and a few friends, cast about for a way to save the old veteran. They found a rescuer in Daniel McCracken, vice president of the Association for Computing Machinery. McCracken paid $318 for the analyzer, topping the only other offer of $100 from a Los Angeles junk company.



Thus reprieved, the machine was sold to the Smithsonian Institution, who agreed to add the differential analyzer to its collection of pioneer computers.

The machine's gears, integrators and plotters are now in Washington, where the Smithsonian's Division of Mathematics plans to reassemble them for a future exhibit on the history of computing.

No More School Daze

Although the number of students attending schools has and will continue to decline in the future, the need and demand for computer systems to arrange bus schedules will grow.

New schools continue to be built because of the shifting population, and school desegregation requires students to be enrolled in in different schools within the same area.

A new software package by NCR Corporation helps schools cope with the time-consuming and increasingly expensive job of scheduling bus transportation.

Advantages of the system include reduced time spent in designing and scheduling bus routes, cost savings in buses and fuel, easier route changes and creation of special reports required for funds from state governments.

The School Bus Scheduling System can be used either with NCR Century or N-mode 8000 series computers. It complements other NCR standard packages for the educational field including budgetary, student record-keeping and learning systems.

A student file is used in conjunction with a system-generated map grid file of all possible bus stops to ensure the best possible routing. The system also prints letters which are sent to the students' homes advising them of the bus schedule.

To qualify for state reimbursement of transportation costs some states require a report showing the shortest actual mileage from each student's bus stop to the school. The system provides the report.

The software package sells for $2000 or $65 a month under a licensing arrangement.

Illustration by Casserine Toussaint

Anchors Aweigh

Shipbuilders at the Bath Iron Works in Maine use their computer and a Material Control System to help manage the flow of material used for shipbuilding, resulting in faster and less costly construction.

From initial design, to the purchase of parts, to the use of those parts by production tradesmen, the system helps the shipbuilding process. Using TV-like terminals linked to a central IBM computer, shipyard workers can instantly retrieve up-to-date information about the availability of material to meet work requirements.

The computer, an IBM System/370, helps identify surplus or shortages among 40,000 inventory items which fill 300,000 individual material requirements. If production schedules change, the computer reschedules the parts timetable and, at the appropriate time, prints out requisitions for additional items.

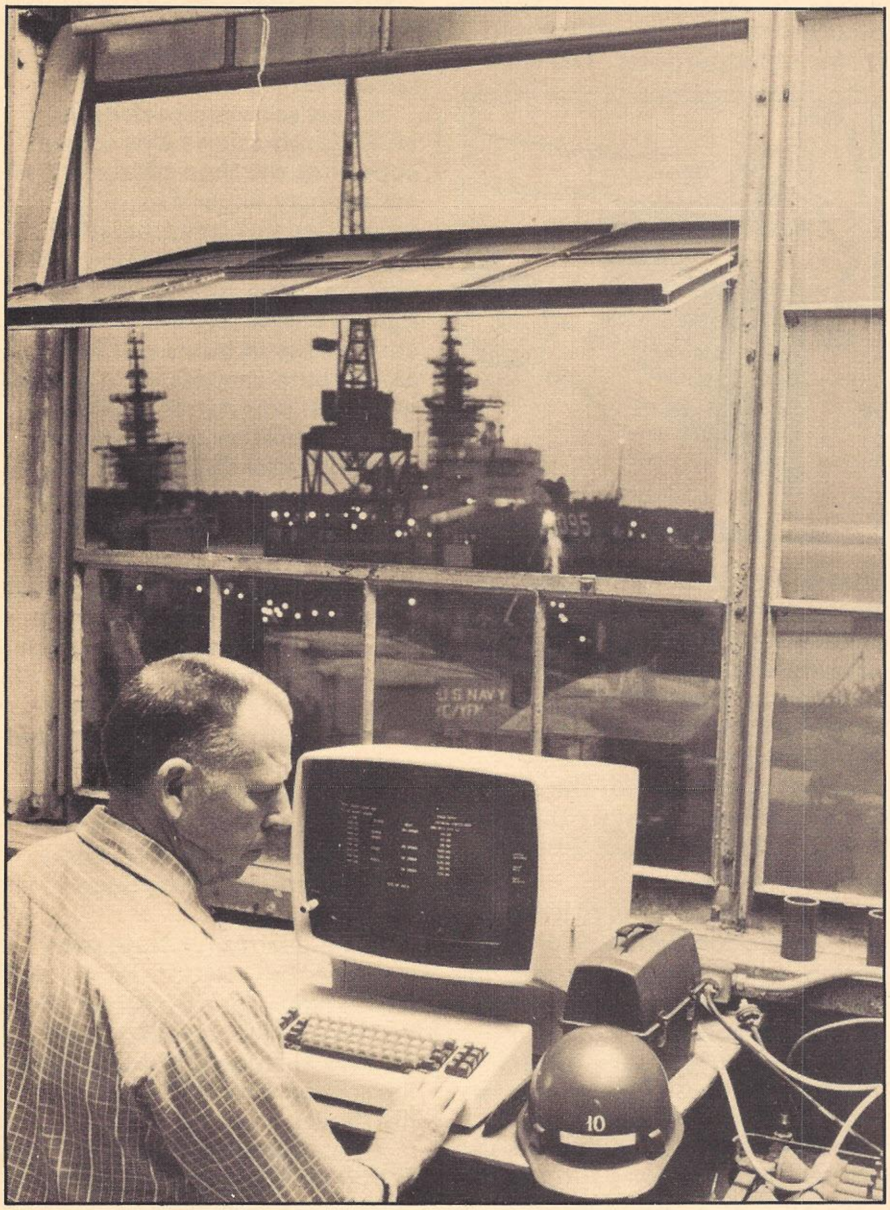
Inventory for the simultaneous construction of Navy and commercial ships, in addition to repair and modernization work in the shipyard, are also controlled by the system.

Before the automated system, raw material was tracked manually as it moved in and out of inventory and through the construction process — a costly, time consuming and administratively complicated procedure.

Other things being equal, cost savings can be passed on to the customer in the bid price of the ship. Specifically, the system allows BIW to provide the timely, accurate information necessary to develop winning bids for new work.

Principal raw materials handled by the computer are steel, cable, pipe, valves, fittings and other non-fabricated parts.

Computer ahoy, maties!



Reference Guide

The Home Computer Guide promises to be a valuable reference work for hobbyists. Now in the planning stages, the book will contain a trade survey of projected home computer sales as well as listing of microcomputer-related firms by product or service and by zip code, the publishers say.

The guide will also include a trade directory of manufacturers, distributors, retailers, consultants, clubs, publishers and suppliers in the home, small business and hobby computer field.

For more information, contact J & M Associates, Box 8118, Kansas City, MO 64112.

Computer Handbook

Newcomers to the microcomputer field may find some valuable information in *The Home Computer Handbook*, though much of the information will be old hat to old timers.

Written by Edwin Schlossberg, John Brockman and Lyn Horton, the book includes product photos and a buyer's guide to home computers, a glossary and a brief history of computers as well as introductory chapters on the potential of personal computing.

The authors previously collaborated on several pocket calculator game books.

Published by Bantam, *The Home Computer Handbook* contains 256 pages plus a 32-page photo insert. Price is $2.95.

"Home" vs. "Hobby"?

In a study on "Home Computers", a Massachusetts-based computer and electronics industry consulting firm, Venture Development Corporation, arrived at a conclusion that many of us realized a long time ago — a true "home computer revolution" won't occur until two conditions are met: widespread consumer appreciation of the computer's capabilities and range of applications, and awareness of the need for these capabilities consistent with the cost of implementing them in the home.

The firm believes these conditions will require development of a new kind of machine — a true "home computer" — which differs from the presently-marketed "hobby computer". VDC defines "home computers" (as opposed to "hobby computers") as fully assembled units with peripherals and power supply, a selection of application programs, a sophisticated operating system and detailed documentation.

VDC expects this "home computer" market to emerge in 1978 as "machines with these characteristics begin to be marketed". They also predict unit sales will increase 250% during the year, then continue to grow 100% per year through 1981.

Since VDC believes today's home computer market consists almost entirely of hobbyists, their study is based on detailed questioning of the hobbyists themselves, as well as manufacturers, suppliers, distributors and other industry experts.

VDC's research shows the typical hobbyist to be young, relatively affluent, well-educated and technically sophisticated in both hardware and software. Surprisingly, more than half of those who called themselves hobbyists don't own a computer.

Through interviews with users, the study also revealed "hobby computerists'" most popular home applications, in order of preference, are games, word processing, data base management, mathematical problem solving, real-time graphics and amateur radio.

As far as VDC can tell, a price decline for "home computers" will not take place over the next few years. Instead, performance improvements (particularly the integration of more I/O devices) will be an important sales factor, and the net result will be a fairly level average price through 1981.

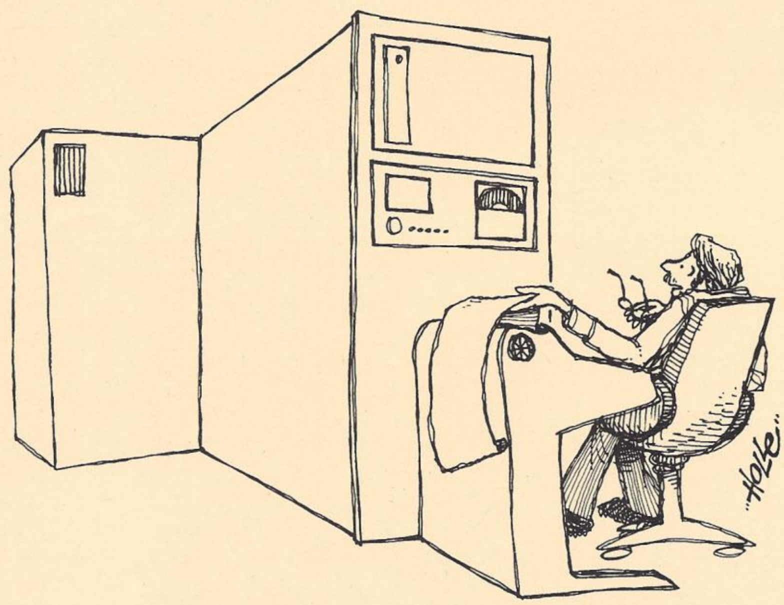
The 168-page study analyzes the history, technology, industry structure, user characteristics, applications, markets, market strategies and future growth prospects of the home computer industry. For more information, contact: Edward A. Ross, Senior Consultant, Venture Development Corporation, One Washington St., Wellesley, MA 02181.

In the cavern of the brain

"The Purposive Brain " (MIT Press), a new book by Ragnar Granit, 1967 Nobel Prize Winner in Physiology or Medicine (with Wald and Hartline) examines the biological purpose and function of the human brain and hints at its relationship to the computer. "Purposive" is a word Granit applies to those actions of the brain that have a definite purpose. As an example of his use of the word "purposive" he quotes, in part, another physiologist, C. S. Sherrington, who said: "The dog not only walks, but it walks to greet its master. The dog's brain alters the character of the walk from one of generality of purpose to one of specific purpose." The "why?" question of nervous system action becomes as relevant as the "how" question.

Granit's work explains visual perception and motor control, the two major functions of the brain that link it to a computer's input and output. The book rekindles the same question that has haunted philosphers since the time of Aristotle: "the purpose of life." You don't have to be a brain surgeon to understand the book or to read about current developments in brain research.

A discussion of the brain is usually difficult to treat in less



"But— they don't **make** Star Wars T-shirts for a computer your size."

than the usual, plodding text-style of writing. But Granit manages it by writing for a living-room gathering rather than for a classroom audience. The book emerges chatty, readable and interesting. But that friendly approach does not detract from the complexity of its subject.

The author explains many natural phenomena such as how evolution develops, how moths mutate from light colors to dark so that they can better blend with environment, how the tongue first served only as a taste organ before man discovered it could be used in speech, and how the dog uses its tongue to get rid of body heat — all of these (and more) acting in response to the purposiveness of the brain.

Other fascinating examples that Granit presents are stereoscopic vision, adaptation, the shifting gaze of the eye. Some of the fields he wanders in, besides evolution, are purpose-and-chance of developement, brain-cell significance, consciousness, perception and psychophysics, and current ideas on brain control of movement.

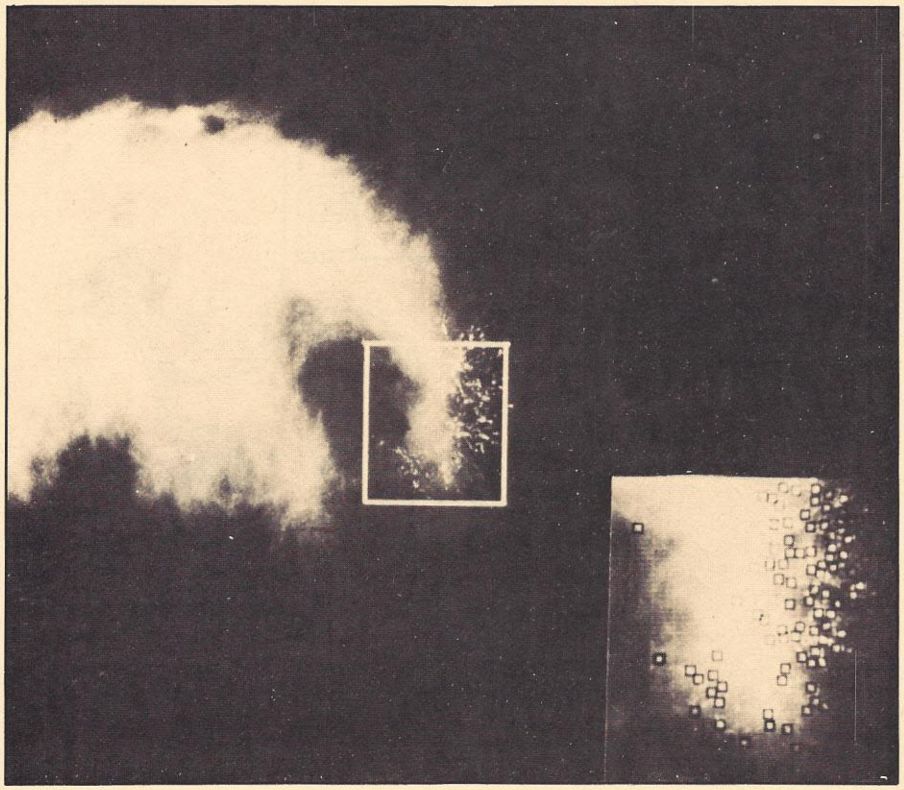
Of particular interest to the computerist is Granit's section on Redundancy. (Read last month's article "The Brain and The Machine.")

Cancer Detection and Computers

In its early stages, breast cancer appears in radiographs as pencil-point-sized patches. Medical specialists identify these incipient tumors by eye — a very tiring task which hardly anyone can perform with reliability for long periods of time. Furthermore, radiographs are often not clear enough for visual examination, causing extra radiation doses and mental stress for the women who must undergo additional testing.

Now, a computer analysis technique developed by Philips Research Laboratory in Hamburg, Germany, eliminates these disadvantages. The Philips method saves doctors from the tiring, routine work by detecting patches of cancer barely visible to the eye.

Thus, women who undergo examination receive as small a radiation load as possible.



Born to Blush

Fumble-fingered computer operators and disk librarians, beware. Shockwatch is watching you. This .3-ounce device, mounted in the disk-pack handle-well, turns red when it detects jolts strong enough to damage the disk.

According to Bob Foley, president of Detectors, Inc., the device's manufacturer, accelerations, jolts, equal to 150Gs (150 times the earth's gravitational pull) will damage a disk "94.7 percent of the time". While 150Gs sounds like a pretty huge wallop, Foley equates it to dropping a disk from a height of 2 feet. In this case, damaging acceleration (deceleration, actually) comes from the interaction of disk with floor.

However, Foley lists many more perils, besides onrushing floors, faced by often defenseless disk cartridges, including "knocking, banging against the drive and even mounting incorrectly."

In explaining how Shockwatch works, Foley says, "Think about a thermometer. When you shake the thermometer, you break the surface tension of the mercury (or alcohol) within the inside tube, and the fluid shakes down. We use the same principle. When a liquid is inserted within a very precise tube, an exact amount of impact is going to be required to dislodge the fluid."

When dislodged, Shockwatch's red liquid, normally hidden, moves into view through a narrow, diamond-bit-drilled channel, causing a noticeable blush. For 150G devices the capillary channel measures .047 inch in diameter. Narrower-channeled Shockwatches require higher G-forces to move the liquid and are, therefore, mounted on equipment more resistant to shock.

Shockwatches currently come in four sensitivity ranges, based on G-forces damaging to protected devices:

- 10 to 35Gs — highly sensitive, for finely-calibrated electronic test equipment.
- 75 to 100Gs — for multi-platter disk packs.

• 150Gs — for 2315 and 5440-type cartridges.
• 200Gs — least sensitive, custom developed for one company to monitor head shaking within Winchester-technology drives.

Detectors, Inc., developed Shockwatch, Foley says, "after an executive of the Burroughs Corp. voiced some real misgivings about the amount of money they were spending repairing and refitting heads on front-load disk drives." Foley then assembled " a number of engineers" to act on Burrough's challenge to build a protection device.

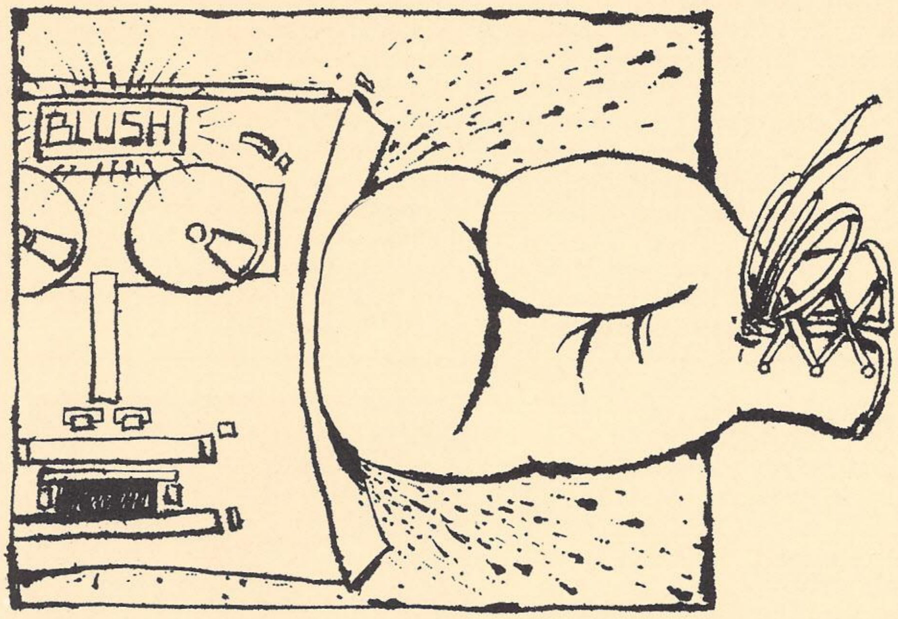
"All that came to no avail," Foley laments, "but a man I had working in my own facility, who was our engineering VP, actually developed this device in his own kitchen. He came in one Monday morning and said, 'Do you think this will do it?'

"He got the idea when driving along after a brief rainstorm. When he hit his brakes, those little droplets of water at the top of his windshield broke away and flowed down the window." Foley's VP reportedly realized that surface tension holding water to the glass broke when the car decelerated . . . and Shockwatch was born.

Burroughs tested Foley's product by affixing Shockwatches to disk cartridges before shipping. "They (Burroughs) gave up keeping statistics," Foley says, "but the controlled shipping produced two devices that had activated. Both disks were returned to the factory and tested." If used, Foley explains, both disks would have caused "a head crash".

Single-user prices start at $6 per Shockwatch, dropping to $4.50 each when purchased in large quantities. "OEM prices we don't like to quote, frankly, but they are considerably lower, depending on volume," Foley adds.

Detectors, Inc., is a division of Media Recovery, Inc., 2550 Electronic La., 75229. — *Dave Powell*



You Don't Say

What's 12″x 10.5″x 5″, weighs 7 lbs. and speaks only when spoken to? "Mike", a microprocessor-based system able to learn and recognize up to 16 user-designated words or phrases spoken by an operator who has "trained" him. Upon recognition, Mike can output an ASCII signal, drive a machine, control an instrument, or access a local or remote data base.

Say a word like COMPUTERS, for example, and Mike will respond by searching the files of the *New York Times* Information Bank some 200 miles away. Seconds later, abstracts of dozens of stories on data processing equipment will appear on a CRT display.

Mike, manufactured by Centigram Corporation, can be trained to recognize a voice in less than two minutes — as long as each of the 16 pre-selected words in his vocabulary is repeated two or three times.

Mike "learns" each word by generating a pattern from the sound waves and storing that pattern in memory. Each word must be repeated, because no one says the same word twice in precisely the same way.

After learning his vocabulary, Mike compares the patterns of each word it hears against those stored in memory. When an acceptable "match" of patterns is found, Mike transmits to the *New York Times* Information Bank a string of characters associated with that word. These characters are transmitted just as they would be if typed on a computer terminal.

The messages retrieve abstracts or summaries of stories on pertinent conference topics from the more than 70 daily, weekly and monthly periodicals included in the data bank.

Presently, Mike's "search" command words are ANTITRUST, AT&T, CARTER (which (which calls for abstracts of stories mentioning President Jimmy Carter after Jan. 1, 1977), COMPUTERS, IBM, MODELS, NEWS (and news media), RATES, SATELLITES and TELEPHONE. Additional "control" words in Mike's vocabulary condition the terminal to learn or to listen, or to page backward and forward through the abstracts displayed.

Search words can be chained together, allowing, for example, mention of both IBM and AT&T, or antitrust proceedings involving either or both corporations.

Mike really represents a glimpse into the future and is by no means a toy. Computer terminals able to recognize human

Illustration by Casserine Toussaint

speech are currently in use in dozens of applications where the entry of computer data on terminal keyboards is either impractical or impossible. Providing invaluable assistance to the handicapped is only one possibility for direct person-to-computer communication. With machines able to talk, as well as machines able to listen and learn, the possibilities are countless.

Other voice recognition terminals currently available feature vocabularies of 50 words or more. Mike is designed to complement these devices, opening up new applications areas. He is also smaller, and therefore more portable, costing only a third as much as the more powerful systems.

Presently, development efforts are underway at Centigram and other companies on terminals able to recognize continuous speech rather than discrete words.

Today, a user with a terminal — whether a printer terminal, a CRT terminal or a voice recognition system such as Mike — need only dial a local telephone number to "tie-in" to resources such as those at the Information Bank.

These files are already accessed regularly by public affairs and other departments within large corporations as well as by libraries, law firms and a host of government agencies.

There is even a terminal at the White House. President Carter's staff used the Information Bank during his campaign for a review of potential Vice Presidential candidates. They searched for each candidate's name in connection with each of a number of platform issues. Fritz who?

Take It Along

You know the old saying — you can't take it with you — but now you can, if it's your micro you're concerned with. There's a new portable microcomputer contained within a briefcase that lets you stay together — wherever you go.

Developed by Adaptive Systems, Inc., the unit operates for 8 hours using a self-contained battery with charger. Data can be stored up to one year in the standby mode.

The system has two 40-key keyboards, an 8-digit LED display and other control switches. Clocking speed is 4 MHz with a 12-bit word size. PDP-8 compatible, most instructions are single-cycle execute.

Applications are expected in vehicular systems, unreliable power sources, data collection in remote areas, on-site data reduction and remote instrumentation. The complete system can be carried on-board an airplane, train or car and data can later be dumped to a larger computer.

Minor variations in design can be accommodated. Price, with no software included, ranges from \$1K to \$3K, depending on memory size, modifications and quantities.

For more information contact: Adaptive Systems, Inc., P.O. Box 1481, Pompano Beach, FL 33061.

Chip Set

Monopoly fans, look out. Video games are on the move. A new chip set for video games from Signetics Corporation offers designers higher resolution and greater programmable flexibility while reducing chip count.

Based on Signetics' 2636 Programmable Video Interface (PVI), the system requires only five integrated circuits in its simplest configuration, reducing production costs for both home TV and coin-operated video games.

The four permanent ICs include the PVI, the 2622 TV synchronizing unit, the NE549 color generator and the CPU — a Signetics 2650 microprocessor. The fifth unit is a plug-in ROM cartridge, which contains the game programs.

Because the system is microprocessor-based and game descriptions originate outside the permanent hardware, new games can be implemented in software for the ROM cartridge without altering the 4-chip PVI module.

ROM size depends on game complexity and can be up to 64K bits.

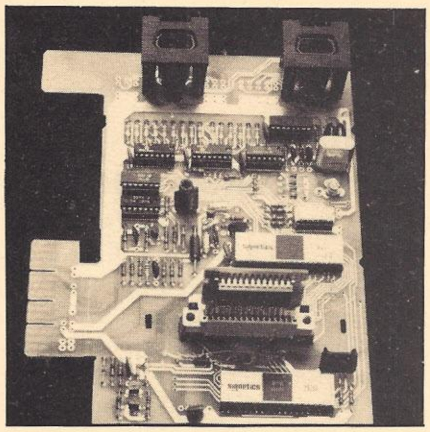
The PVI can generate up to 80 separate images from four object modules, in up to 16 colors. Other built-in capabilities include eight digits of numerical data for scoring, eight color choices for screen background and programmable music capability — up to 255 frequencies encompassing a three octave range.

The system's object-oriented approach to visualization eliminates the need for large RAM capability. Descriptions of the game pieces are stored in ROM and fed to object descriptors in the PVI, where the information can be used to define size, shape, position and color. The microprocessor manipulates the described object according to input from the players (via potentiometers and/or switches), and game programs contained in the ROM cartridge.

By contrast, in a RAM-intensive approach every possible position on the screen for each ob-

ject must always be represented by one or more bits of memory. The resolution provided by the single 2636 PVI would require 40 or more chips in a traditional RAM-intensive system to constantly map the screen with all possible image positions.

The 2636 PVI is a bus-oriented device with the microprocessor



address and data buses entering the PVI and accessing the major functional blocks. The microprocessor selects the block it wishes to communicate with via the address bus, while the actual information is presented to or received from the selected block on the data bus. The object and background video patterns are stored digitally in a RAM area on the PVI chip, where they can be accessed by the microprocessor and load the vertical and horizontal counts to generate the specified video at the proper location. The PVI also contains a digital sound block (a square wave generator) for programmable tones, and a ROM Chip Enable Block to reduce the system's component count.

The PVI can display 200 lines of video information with 280-nanosecond horizontal resolution. With 80 bits of RAM dedicated to object shape description, game designers can use this high resolution to program finely detailed and stylized game pieces. Each of four pieces can be duplicated in several sizes up to 20 times, depending on the complexity of the game.

Background patterns can be displayed with up to 320 distinct vertical bars in four different widths, while the color generator allows selection of up to eight background hues and 16 object (game piece) hues.

The games set is in production with samples now available.

For further information contact Signetics, P.O. Box 9052, 811 East Arques Ave., Sunnyvale, CA 94086.

Catching Crooks

Crooks trying to cash counterfeit checks are going to get caught — if NCR Corporation has anything to say about it. They've developed a printing technique that causes the word "COPY" to appear on the face of checks and valuable documents copied by a color copier.

Their technique, "Stop-a-Copy", provides, in effect, a built-in automatic alarm system on the face of each check.

On the original check the word "COPY" blends in with the colored background, not clearly visible to the unaided eye. However, on a color copier reproduction, the background fades, leaving the word "COPY" visible to innocent bankers or shop owners who might otherwise cash the counterfeit check.

In addition, NCR prints checks with patterns or designs on the back, visible to the eye but which cannot be copied by a color copier. A message on the face of the check notes a "genuine check" has printing on the back.

Should ink eradicator be used in an attempt to alter the amount or the signature, a third optional feature causes the word "Stop" to to appear where the eradicator has been applied.

NCR has little evidence that the advent of color copiers has caused any increase in the counterfeiting of checks. However, they note the high potential for fraud given the billions of checks written each year and the 15 to 20 separate steps each check goes through in the clearing process, and an increase in that potential as a result of color copiers coming onto the market.

You can use "Stop-a-Copy" with payroll checks, money orders, voucher checks, gift certificates, dividend checks and cashier checks. The potential uses are limited only by your imagination.

PET Training

"Getting Started with Your PET", a beginner's workbook from TIS, supplements documentation provided by Commodore. The descriptive text is laced with step-by-step, detailed exercises including the expected PET responses.

If you are already an expert on your PET, "Getting Started" can serve as a guide for other members of your family.

Workbooks on advanced topics, covering string handling, arrays and loopings, graphics, cursor control, PEEK and POKE memory, programmed cassette I/O, real-time clock, linkage to assembly language subroutines and subroutine nesting, are available.

TIS also provides PET applications as source listings or cassettes with operating instructions, theory of operations description and performance time and space limitations.

For more information, write to TIS, P.O. Box 921, Los Alamos, NM 87544.

NCC Sells Out

Exhibit space for the 1978, June 5-8, National Computer Conference has been grabbed up and is now sold out. NCC's sponsor, The American Federation of Information Processing Societies, Inc., (AFIPS) notes that 330 organizations have reserved 1382 booths — the largest exhibit of computer hardware, software systems and services ever.

If you find a way to make it to the Anaheim, CA, conference, stop by and visit us at the *Personal Computing* booth.

On Target

It may not be a matter of life or death, but you certainly don't want to announce a winner prematurely — only to later find out you were mistaken. And in the past, because of the time involved with collecting and manually tabulating ballots, nearly 300 sports writers and sportscasters who select the National Basketball Association All-Star game's Most Valuable Player have had to cast their votes at mid-point in the fourth quarter.

But this year, they were able to wait until the final buzzer to vote. And it's a good thing because, with only 5:10 remaining on the clock, the East All-Stars, having trailed for the entire game, took over the lead and held onto it for a win over the West.

Less than one minute after the final buzzer, Honeywell computers proclaimed Randy Smith of the winning East team MVP. Smith received the NBA's MVP trophy and keys to a new Chevrolet.

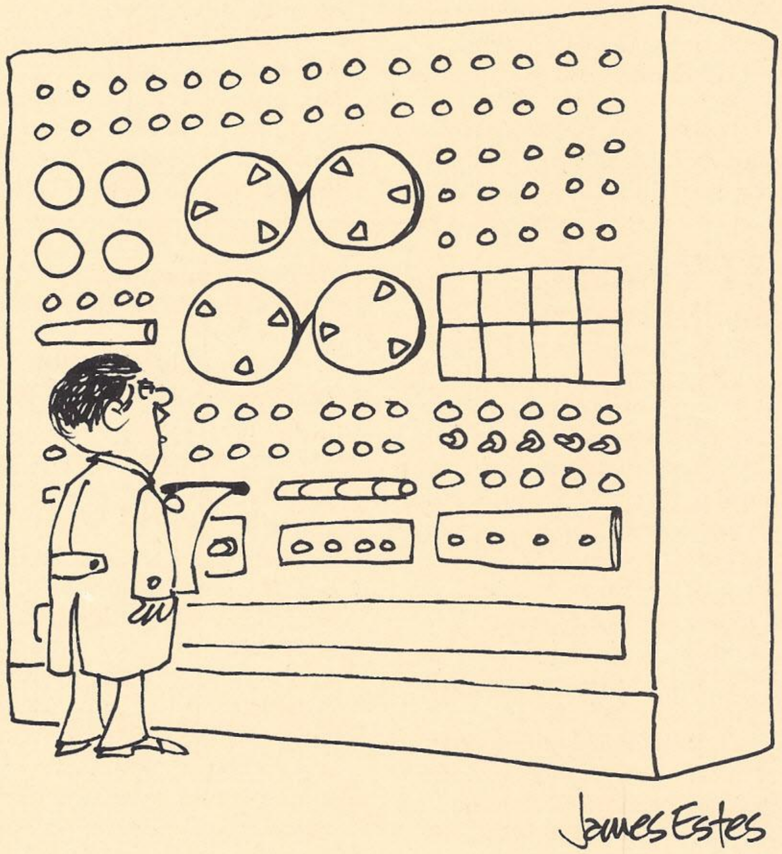
Historically, the MVP has been a member of the winning team. Without the computerized tabulation of the votes, the ballots would have been collected while the West was ahead — and a lot of sports writers might have nominated a member of the team that would later lose the game.

In the Atlanta arena, 20 computer terminals in five areas were situated near voting sports writers and sportscasters. At the one-minute-remaining point and at the final buzzer volunteers collected the ballots and rushed them to the nearest terminals.

Each terminal was connected by dial-up phone lines to Honeywell's data center in Atlanta. (As a backup system, each terminal was also tied to the firm's data center in Phoenix.)

Entering the votes into the terminal took just over a minute, and within a few seconds after entering the last vote, the winner's name flashed on a 24-inch visual-display terminal which CBS Sports had on-camera.

Maybe whether you win or lose does determine how you played the game?!. . .



"OH, ALL RIGHT, THEN... SIMON SAYS 'COMPUTE!'"

Consumer Electronics Show

The 1978 International Summer Consumer Electronics show will be held June 11 to 14 at the McCormick Place/McCormick Inn/Pick Congress Hotel in Chicago. More than 800 exhibitors will use over 400,000 square feet. Show managers estimate attendance at 50,000. The show is sponsored by the Electronic Industries Association Consumer Electronics Group. The 1978 exhibition will be the largest in the show's twelve year history.

If you drop by the show, be sure to stop at the *Personal Computing* booth to say hello.

Televerket Anyone?

Televerket, the Norwegian Telephone and Telegraph Company, hopes to improve customer service and contribute to more efficent and economical operations by employing a new Sperry Univac computer.

When it becomes operational in December '78, the system at Televerket's facilities in Oslo will be used to provide various information services for telephone customers, produce telephone directories and furnish a control and information system enabling Televerket to use more economical techniques.

Several hundred display terminals located in local telephone offices throughout Norway will be connected to the computer, enabling district authorities to access information in the computer's data base within a few seconds.

Business Ed

More and more education today emphasizes practical application for the business world. In response the Olivetti Corporation of America, in conjunction with business educators, has designed a programmed instruction course on electronic bookkeeping at the secondary, vocational school and community college levels. This business education package runs on the Olivetti A4, an electronic numerical accounting machine.

The course covers the language, logic and procedures common in today's automated offices. Students learn to perform debit and credit posting, accounts payable, accounts receivable, general ledger accounting and billing.

Students progress at their own pace with teachers serving as "managers" rather than instructors, to further reflect the working world.

A student text and teacher's guide include three activity units. Part I explains operation of electronic computing machines used for posting accounts receivable records in the business office; Part II covers processing invoices and bills while teaching pupils the procedure used to assure correct billing completion; and Part III discusses the basic characteristics of electronic data processing systems as well as the capabilities of the system as an electronic printing calculator.

A personal review and answer key for self-checking follows each unit.

The A4 unit can be used as a four-function electronic printing calculator in a wide range of applications including an input unit to an integrated data processing system. Because it is programmable, it's an effective tool to interest students in the data processing field.

The unit consists of a central unit with numeric and function keyboards, a control console, 16 characters-per-second print unit, program drum read unit and an interface adapter for connection to peripheral units. Standard paper controls include a rear feed for journals and a front feed for ledger cards and single sheets. A lighted display guides the user at each step of operation.

The complete Business Education Program, including hardware, curriculum, manuals and software, is available starting at $2,945. Olivetti provides user training and support as well as service.

For more information, contact Olivetti Corporation of America, Educational Systems, 500 Park Avenue, New York, NY 10022.

Rain, rain, go away

Every telecommunications expert talks about the weather and its effects, and finally one has done something about it.

Rain weakens radio signals transmitted on microwave terrestrial systems, so Evan J. Dutton, an Office of Telecommunications physicist, wrote a computer program, PRED 77, that predicts the rainfall rate in a given area of the U.S. and the extent to which this rainfall will weaken radio signals. The program is similar to one developed earlier for Europe.

If you're concerned with the amount of precipitation Mother Nature has in store for your area, you can own your very own copy of Dutton's report — OT Report 77-134, *Precipitation Variability in the U.S.A. for Microwave Terrestrial System Design.* Order from the National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161. ($7.25) Mention accession number AD A049041.

In-house system

Up to now the city of Sacramento, California, used a computer owned by Sacramento county to run its computer programs. Now they plan to install their own system and save more than $600,000 in computer costs over the next six years. A total conversion of existing programs with the Sperry Univac 90/70 system programs is projected for May.

Use of their own in-house com-

puter will provide the city with the ability to control, schedule, plan and organize the use of computer resources. In addition, it will provide privacy and security for its users.

The computer programs will encompass a wide variety of applications including bi-weekly payrolls, budgeting, systems for the police and fire departments, pension plan administration, vehicle maintenance, business, animal and bicycle licensing, and utility billing.

Sacramento is just one of many, many cities turning to computers for economic reasons and quantity and quality output.

Bank On It

The European Asian Bank has become the first bank in Hong Kong to install a NCR I-8250 small business system, a computer featuring multiprogramming and interactive direct processing.

Heinz Wiens, chief manager of the Hong Kong branch, said the interactive on-line system will upgrade the bank's information processing capability and streamline its reporting to both Hong Kong and German banking regulatory agencies.

Based in Hamburg, West Germany, the European Asian Bank has branches in Jakarta, Karachi, Kuala Lumpur, Manila and Singapore, with additional branches planned in Bangkok and Seoul.

Down on the farm

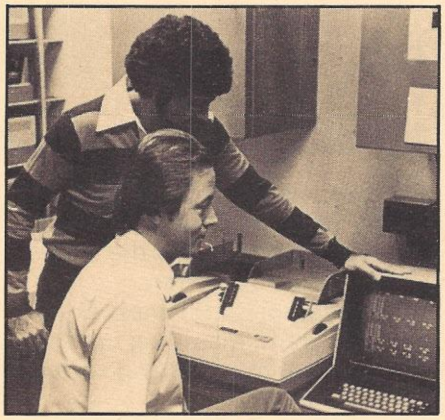
While the organ plays, the merry-go-round spins and the arcades gobble your loose change, three desk-top microcomputers work silently behind the scenes to keep Knott's Berry Farm running smoothly.

They're integral to the Construction and Maintenance Division's activities in scheduling and expediting jobs in progress, lighting control and air conditioning control for energy conservation.

The amusement park, which hosts four million visitors a year on its 150 acres, features more than 100 rides and attractions, eating places and shops.

Three MITS/Altair 8800 microcomputers, which provide the central control for the division's systems, were built from kits by the Farm's electronics department. Each of the central processors is dedicated to a specific application (work order scheduling, lighting, and air conditioning) but any of the terminals and printers can be switched as needed to any of the three applications.

Computer system hardware includes three Ann Arbor terminals, three 110-cps Okidata line printers, and three disk drives. The work order scheduling system uses 40K bytes of mainframe memory; the lighting and air conditioning systems use 16K and



20K, respectively. Each of the three disk drives uses 300K bytes of floppy disk storage.

Design and implementation of the lighting and air conditioning control portions of the system entailed installation of approximately $75,000 worth of acoustically-coupled transmission and receiving equipment to control 11 separate lighting areas and approximately 50 air conditioning units.

Total cost of the computer hardware for the three-part system — excluding the transmission/receiving equipment — was $21,800.

The work order expediting system schedules the Construction and Maintenance Division's workload based on due dates and priorities assigned to the various jobs. When more than one shop is involved in a particular job, the computer coordinates the different shops. A printed listing, showing all jobs in priority order, is furnished periodically to the division's managers.

In the lighting control system, the computer provides centralized control to turn groups of lights throughout the Farm on and off at pre-specified times. "On" and "off" times for special events that require lighting at unscheduled times can be entered into the system. After the special event is over, it's purged from the memory and the system reverts to to its normal schedule.

All of the Farm's lighting, except a few parking lots and other peripheral areas, is under computer control.

Primary purpose of the third computerized system is conserving energy and minimizing the Farm's electric bills through air conditioning control. The system compares actual kilowatts used by the entire Farm at any given moment with a "standard kilowatt" amount designated as a standard for that particular time of day. Whenever the actual kilowatt amount exceeds the standard, air conditioning units cycle off and on at short intervals — usually every ten minutes.

One feature built into the computer system guards against excessive "peak" power usage which could adversely affect the Farm's utility rate structure. In the event that power usage should reach a "maximum kilowatt" figure, the system will continue to cycle as before, but will increase the "off" times by 20 percent.

The Farm plans to add another application to the system in the near future — inventory management — to keep better track of the parts that various shops need. Shop personnel will note use of parts on their daily work cards and the computer will report low inventory levels.

So when you're rocketing down the roller coaster, try to give a moment's thought to the micros behind the scenes.

STRUCTURED PROGRAMMING

BY WILLIAM L. ROBERTS

One of the latest fads among professional programmers is "structured programming". From a beginning about ten years ago, the idea has spread across all types of programming. Proponents of the idea claim it solves all kinds of programming problems from fast implementation to simple maintenance. However, very few people can describe what structured programming is and those who can each describe it differently.

In this article I will describe structured programming as I understand it and use it in my work. I hope my explanation will help you write better programs, or at least understand what structured programming is.

In structured programming, the entire program is built from individual structures – just like the hardware of your computer. A machine is made of several dozen to several hundred thousand chips. If you try to figure out the whole machine at once, the task is mind boggling. You think instead of the CPU. You first decide that it is in the memory; then on a specific board (16K boards, of course – this is a big system); and finally, in a specific module. When designing a system, you first start with the idea that you need a memory, then design the boards, then select the chips. Structured programming allows you to do the same thing with the software. At each structure you define what goes in and what comes out.

In hardware, the structures used are the chips and boards, or combinations of them. The most important things to notice are: 1. The inputs to the chip or board are clearly defined (5+ volts on the line, address lines set) 2. What happens to those inputs are defined (tape will read the next byte) and 3. The outputs are clearly defined (data lines contain the byte read, halt line set).

In software, you can define input and output in a process called HIPO (hierarchy, input, process, output), diagrammed as shown in Figure 1.

Just as the hardware designer would

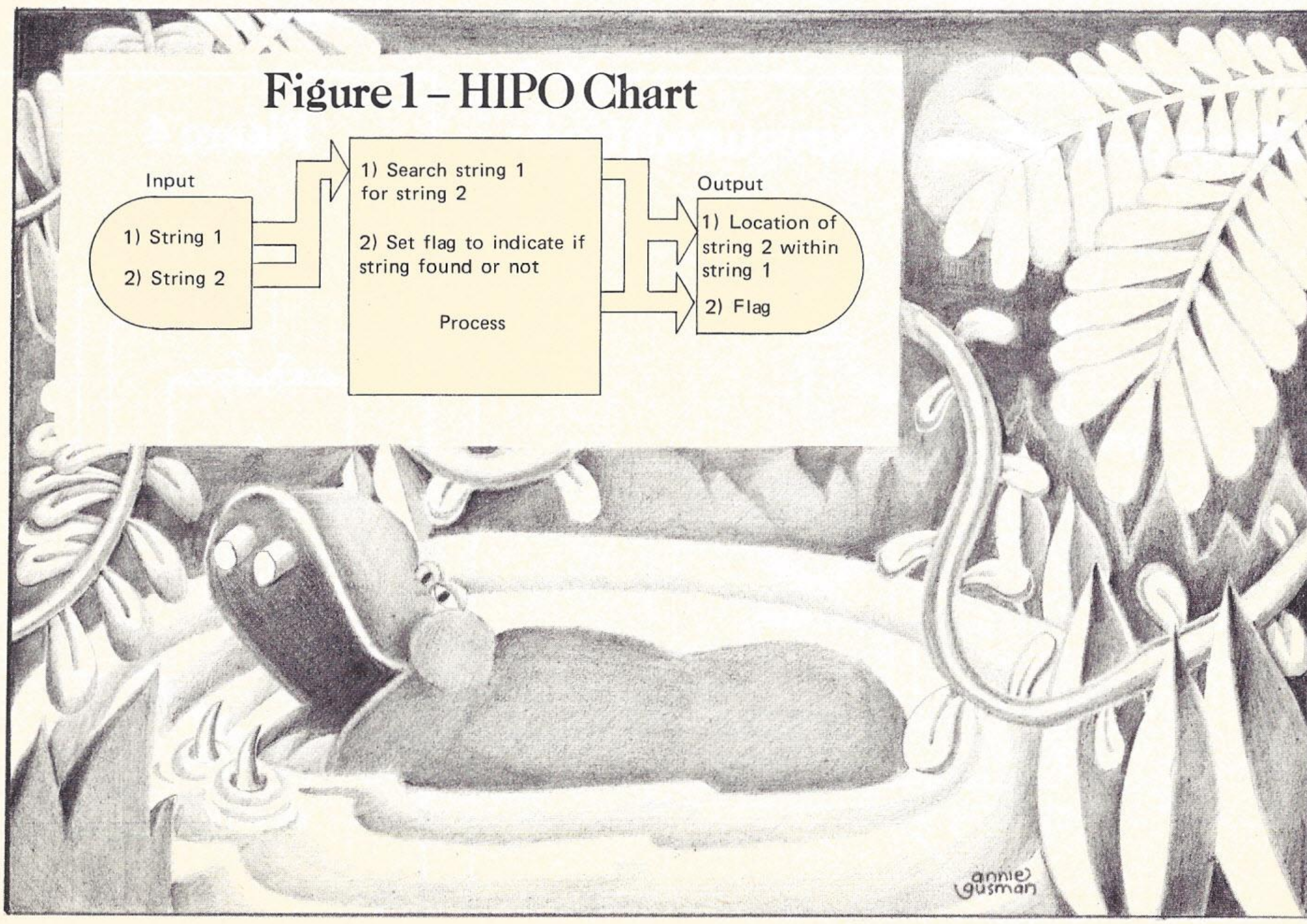
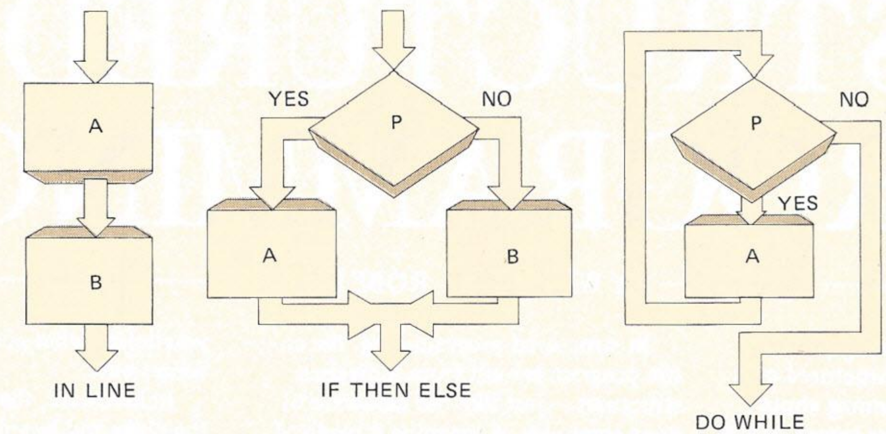


Illustration by Annie Gusman

Figure 2 – Basic Structures



not decide the chips and boards he needs until he knows exactly what the hardware has to do, the software designer should not start coding until he knows exactly what he has to code. It is surprising how many programmers will write the code before they know what they have to do, and end up with a piece of code that does not do what they want. (This is known as a bug.)

Thus, the first rule of structured programming is to draw or write a HIPO for the program (and every smaller unit) you are doing. List every input, tell what the code will do (not how you will do it), and tell every output from the code. If you can't write a HIPO, you are not ready to write the code. If during coding, you discover you need a new input or output, change the HIPO first; then go over all the code you have written to see how it is affected; then change your code. This procedure is not as hard as it seems, since you have the HIPOs for all the code. If you follow the techniques I will give, it will become fairly easy to see the impact of change on code.

What are the structures that the code is built from? To the purist, there are three, shown in Figure 2 in flowchart symbols.

Before I continue, let me give a brief explanation of the symbols. The diamond is a decision box. It asks a question that can be answered in one of two ways (is A > B, is string N$ in string A$). You follow the "yes" path if yes is the answer to the question; otherwise follow the "no" path. The rectangular box describes some process (LET A = B + C, GOSUB 6000, find average value). Note that this process can be one statement from a program, or a description of a complex process

Figure 3 – Inefficient (Structured) Code

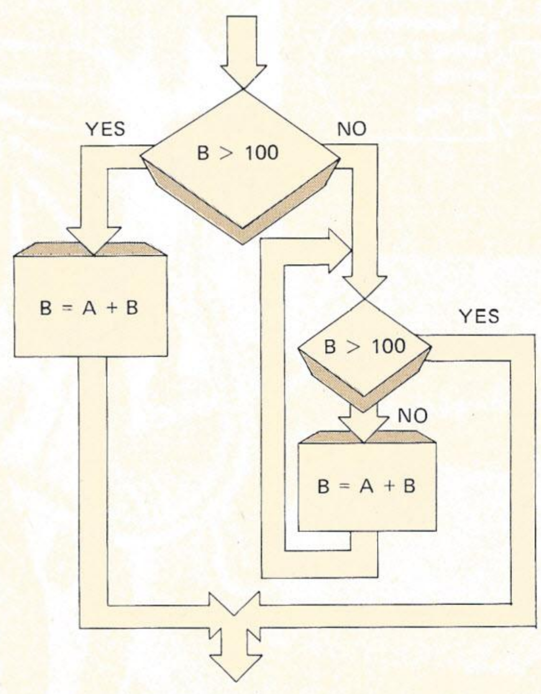
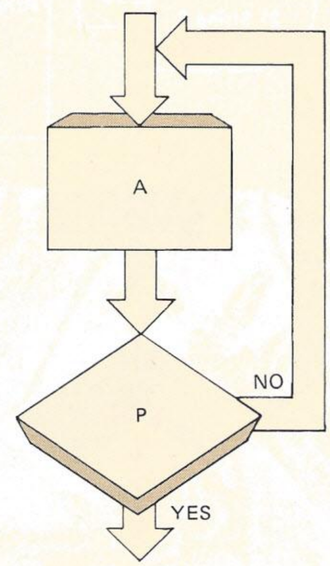


Figure 4



which must itself be described in a HIPO and flowchart.

Now for the structures (see Figure 2). The first one is in-line, where two processes occur one after the other. In our example in Figure 2, notice that we do task A, then task B, then go on. We do not go back to A. Your reaction may be "Whoa, what about a loop?" We'll get to that.

The second structure is the IF THEN ELSE structure. In our example, if P is true we do A, else we do B. No matter which we do, we come back together again afterward. Notice that this procedure ("Take action based on P") can be used in the in-line structure above. The in-line structure is also a process, so we can say IF P THEN A; B ELSE C; D. You must know what goes in at the top, what the process is, and what comes out at the bottom. For any process, there is only one way in, and one way out.

The third structure is the WHILE DO structure – while P is true we do A. If P is not true, we don't do A. Therefore, in A you must do something to change the value of P. Notice, though, that you can only enter at the top and exit at the bottom. In that way you can say what you expect in and what you will give back. (This structure is the loop I mentioned earlier.)

Any program can be written using these structures, but to do so often requires some special and rather weird logic. For example, suppose we want to add A to B until B is greater than 100, and we want to do that addition at least once. Figure 3 shows the flowchart we need. The code for that chart is:

```
1000  IF B > 100 THEN 1100
1010  IF B > 100 THEN 1200
1020  B = B + A
1030  GO TO 1010
1100  B = A + B
1200  . . .
```

This code shows two of the arguments against structured programming. First, it takes more code than unstructured; second, it takes more time to run. I believe that I can solve (or at least ease) both problems by introducing three additional structures, which I will do shortly.

But I would like to offer some rebuttal to the above arguments. First, structured programming makes you stop and think about what you are coding, often making the code shorter and faster than it would have been. Second, debugging is much easier with structured code. Not as many patches (and they require space) are needed. Finally, the space saved and/or time gained avoiding structured is purchased at the cost of more bugs and more difficult maintenance. Of course, when you are out of space, you're out, and must shorten the program. But you can shorten structured code easier than unstructured. So your best bet is start structured, and only abandon it when you have to.

My added structures are shown in Figure 4. The first is the DO - UNTIL structure, which solves the problem mentioned earlier with the WHILE-DO. The corrected flowchart is shown in Figure 5. The code becomes:

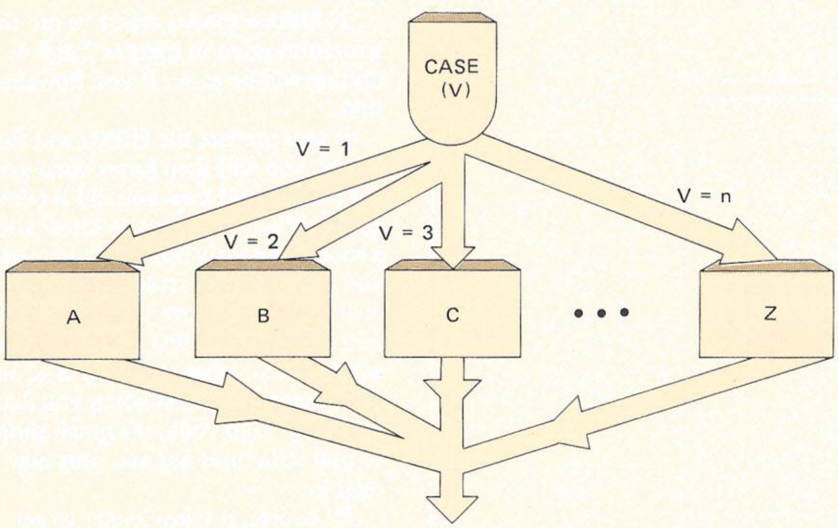
```
1000  B = B + A
1010  If B <= 100 THEN 1000
```

The second structure is LOOP - EXITIF - ENDLOOP, which I introduce with a bit of trepidation. You enter a loop and exit if a certain condition occurs. This structure still meets all the requirements I set up (known conditions in and out; one place in, one place out). As an example, suppose we want to add A to B; and if B is not greater than 100 then add C; then A again until B is greater than 100. Our code becomes:

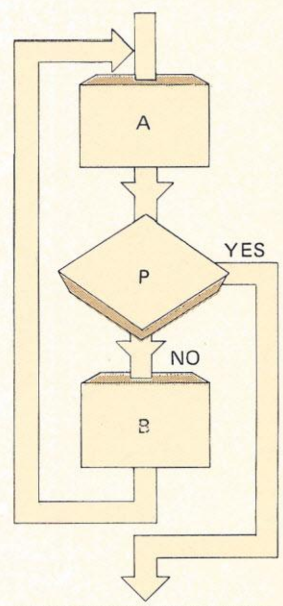
```
1000  B = A + B
1010  IF B > 100 THEN 1100
1020  B = C + B
1030  GO TO 1000
1100  . . .
```

My reason for trepidation is the temptation to add several EXITIFs to the loop.

Additional Structures



CASE



LOOP EXITIF ENDLOOP

Figure 5 - Efficient Structured Code

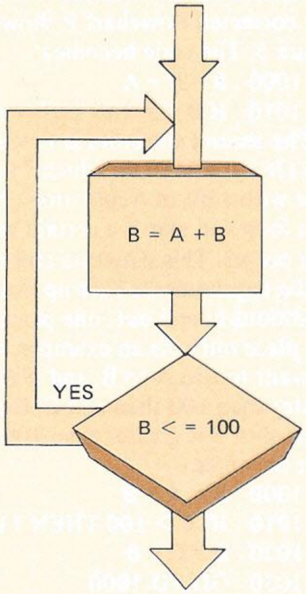


Figure 6 – Loop With Several Exits

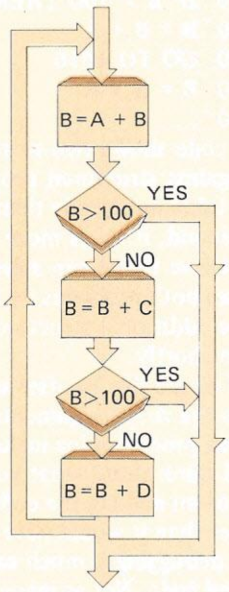
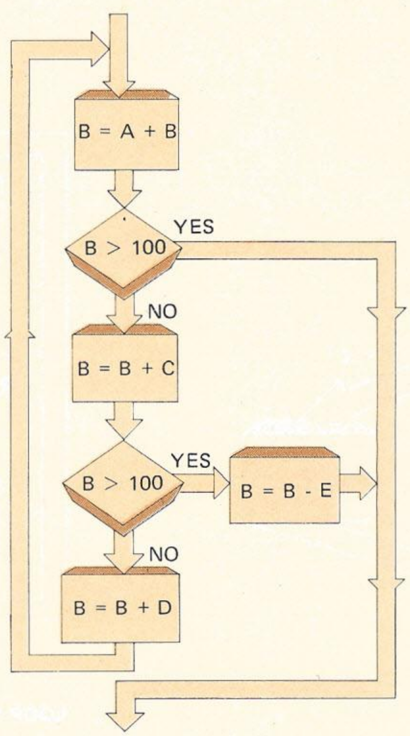


Figure 7 – Loop With Several Exits And Special Handling



For example, given the above situation, let us add D to the list of variables. We will add A, test; then add C, test; then add D. Our new flowchart is shown in Figure 6. As they say in math books, the code is left as an exercise for the reader.

Notice we have violated one of my original requirements for a structure. We do not exit from one place. We do, however, exit *to* one place, so I'll let it go. But this is the only rule I'll change: you must exit to one place (Note: within sub-routines, this exit may be a RETURN statement).

I have one more example in the LOOP, and then I will consider it beaten to death. Suppose in our example in Figure 6, if we exit at the second exit (after adding C), we want to subtract E from B. This procedure gives us the flowchart in Figure 7. I still consider this a structured piece of code because it enters at one place and exits to one place.

The last structure is the CASE structure (see Figure 4). One of the several possible actions is selected depending on P. BASIC often implements CASE structure by 1000 ON P GO TO 1200, 1300, . . .

Remember that after each special action is done, you must come back to one place.

So to wrap up: To apply structured code to your programs, use the following guidelines:

1) For any process, draw a HIPO before you code. List the inputs, the processing done, and the outputs. When you have the HIPO, you know what to code.

2) Make a special effort to use the structures given in Figures 2 and 4. Coding will be easier if you flowchart first.

If you prepare the HIPOs and flowcharts, you will then know what you did (HIPO) and how you did it (flowchart). When you want to know what a routine does six months from now, you don't have to struggle through code which you have forgotten about. Just look at the HIPO and you know what goes in, what the process is, and what comes out. And when you decide to change your code, it's much simpler to pull structures out and plug new ones in.

Of course, it takes longer to get from an idea to code – and code is the fun part. But it's even more fun to make the code run the first time.

The Computer for the Professional

The 8813 was built with you, the professional, in mind. It quickly and easily processes cost estimates, payrolls, accounts, inventory, patient/client records and much more. You can write reports, briefs, and proposals on the 8813's typewriter keyboard, see them on the video screen, and instantly correct, revise, or print them.

Using the 8813, one person can process what would normally require many secretaries, several bookkeepers, and a great deal of *time*. And data storage takes a small fraction of the *space* used by previous methods.

You don't need to learn complicated computer languages. The 8813 understands commands in English. If you want to write your own programs, the 8813 includes a simple computer language, BASIC, that you can master in a few days. The 8813 slashes the professional's overhead. It's a powerful time and money-saving ally. Prices for complete systems including printer start at less than $8,000.

See the 8813 at your local dealer or contact PolyMorphic Systems, 460 Ward Drive, Santa Barbara, California, 93111, (805) 967-0468, for the name of the dealer nearest you.

PolyMorphic Systems

If It's Tuesday.. This

Without proper planning, your summer vacation across country by car could turn into a disaster. You could end up 500 miles from home with two vacation days left and $5 in your pocket.

Or, you could avoid these problems by planning your trip in advance – with your microcomputer.

"Planner", a microcomputer program, lets you calculate how much your trip will cost and the time it will take you to travel to your selected destination(s).

With this program, you'll know before you pull out of the driveway whether you can fit visits to Mexico City, Phoenix, Salt Lake City and Seattle all into your one week vacation. You'll also know whether you can finance your trip with your vacation pay or whether you'll need to break open your piggy bank.

If you're the type who gets carsick on a merry-go-round, acrophobia on a step ladder and nauseous smelling salt water, you can use Planner as a game – compete with friends to see whose trip plan will cost the least money and take the least travel time.

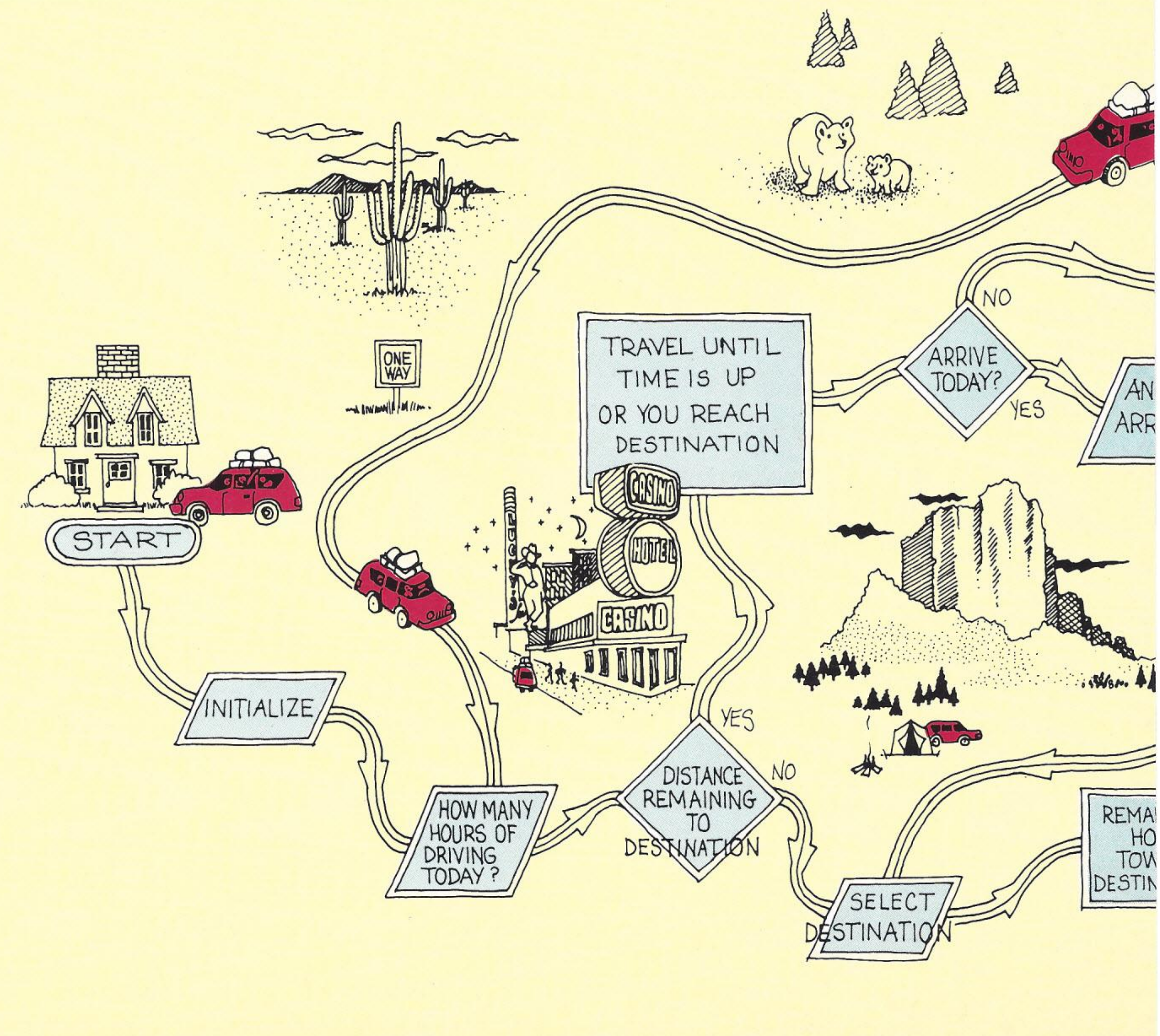
If you're the entreprenurial type, Planner has great potential as a "lemonade" service. Design vacations to suit your friends', neighbors' and relatives' specifications.

Children can even use this program to plan a trip of their own or as a game. The distance table built into the program helps youngsters learn about the geography of the United States.

Planner's cost figures, along with the highway mileage table are based on American Automobile Association (AAA) estimates.

A run of the program provides you with an itinerary for every day of your trip, daily expenses broken down into variable and fixed costs and a projection of the total trip expenses, also broken down into variable and fixed costs.

The program takes into account the following ex-



Must Be Seattle

penses: gas and oil, car maintenance, wear on tires, insurance, taxes, depreciation, motel lodging, food, and sightseeing costs.

Not included in the expenses are the following costs: highway tolls, souvenirs, amusements and a contingency fund (do keep these in mind when planning your budget).

Vacation Expenditures*

Though vacation expenditures depend on personal preferences, regions traveled, and individual means, advance planning can mean the difference between a successful vacation and one hurriedly cut short for lack of funds.

For an automobile vacation (See Sample Run) two people should plan on spending $26 per day for meals (not including alcoholic beverages and tips), $27 per day for lodging and $5 for gas and oil for every 100 miles of travel (for a car averaging 15 miles/per gallon).

Many establishments offer a family plan where you pay only a small additional charge for a third or fourth person occupying a room. Anticipate an average $3 per person, per day charge, and possibly less for children younger than the age limit set by management.

The suggested daily budget for meals and lodging varies depending on locality. In small towns or villages, these costs might be as much as 30% higher.

Remember, winter and summer "resorts" have higher rates during the main season and lower rates during the off-season.

In addition to food, lodging and car operation, don't forget expenditures for souvenirs, amusements, admissions to places of interest, recreation, retail pur-

* **Extracted from "Your Driving Costs", 1977 edition, published by the American Automobile Association, 8111 Gatehouse Rd., Falls Church, VA 22042.*

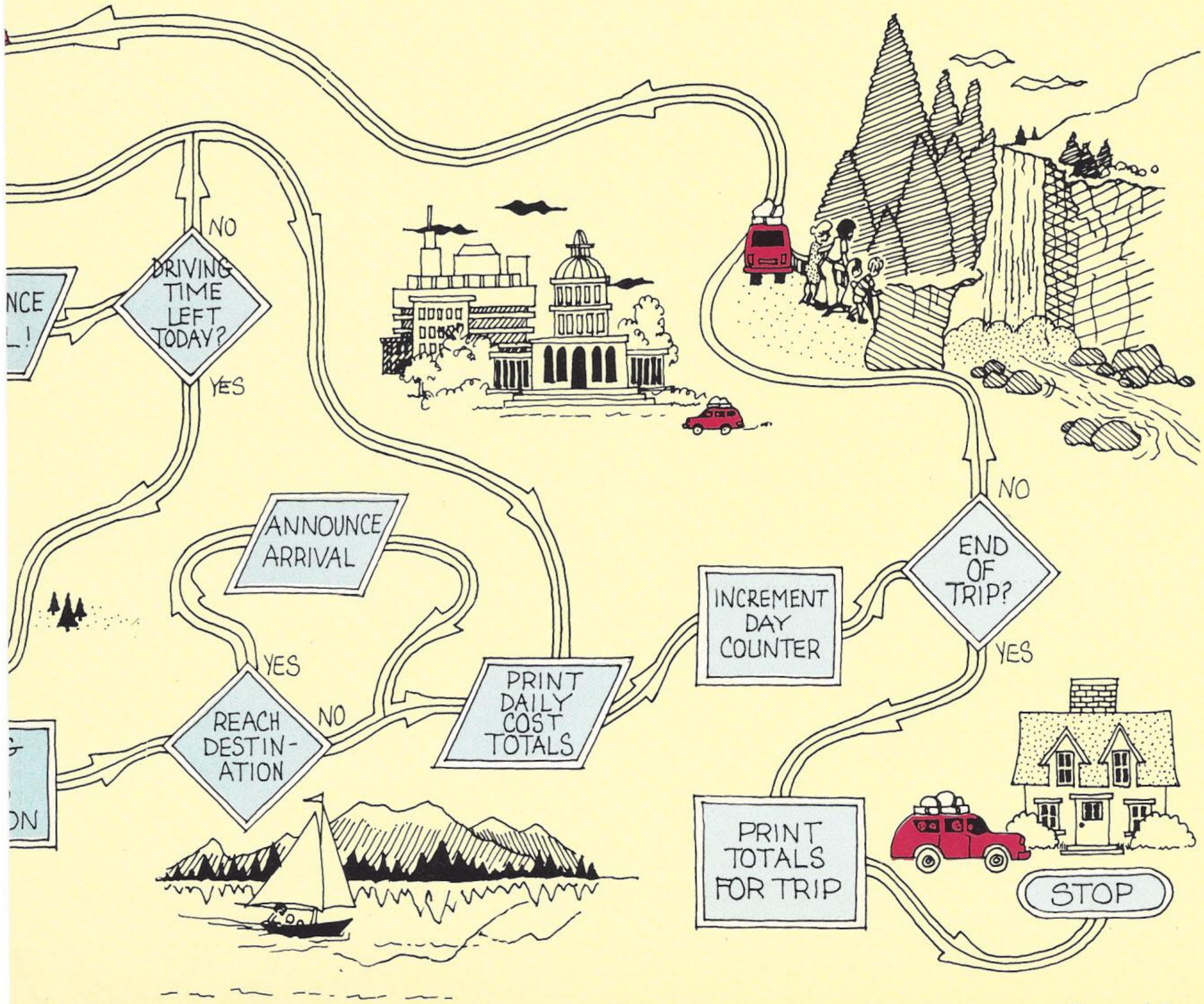


Illustration by Jane Higgins

chases and so on. Road and bridge tolls, depending on the routing, also will be an expense. Include a contingency fund for emergencies in the vacation budget.

Itemizing Car Expenses**

The cost of car ownership breaks down into two categories: variable and fixed.

Variable, or running costs, include gas and oil, maintenance and tires. They are directly related to the number of miles driven, the type of driving (city, flat country, mountains) and how much you spend on service and repairs.

Fixed costs include insurance, license and registration fees, use and property taxes, and depreciation. Fixed costs may vary from car to car and from place to place, but they are established by agencies and business conditions beyond the control of the car owner and change little with the amount or type of driving.

Here is a sample breakdown of variable and fixed costs for one car. These national average cost figures were computed by Runzheimer and Company for a 1977 intermediate-size Chevelle, 8-cylinder (305 cubic inch) Malibu Classic 4-door sedan equipped with standard accessories, automatic transmission, power steering, power disk brakes and radio.

PROGRAM NOTES

The following description of the variables used in Planner should help clarify the program.

"RM" is the variable for "remaining miles to destination." It is set whenever you set a new destination in line 2065. Then it is reduced by your driving until it reaches 0, when a new destination may be entered.

"C1" is the number of the starting city on a particular leg of your trip.

"C2" is the number of the ending city of a particular leg.

"H" is the number of hours left to drive during a day. It is set in line 230 and reduced by driving until you reach a destination or it reaches 0 and you run out of driving time on that day.

"D(I,J)" is the matrix containing the distance table. I and J are the starting and ending cities, and D(I,J) is the distance between them.

"DT" is the total number of days in the trip.

"DC" is the day counter, which increments after each day is over.

"D1" is the number of miles you travel on a particular day.

"MH" is average miles per hour.

"ST" is number of miles of sightseeing travel.

"SC" is sightseeing cost for a day.

"PP" is number of people on the trip.

"RC" is cost per night of lodging.

"D2" is sightseeing plus travel miles.

"C" is used to printout the proper city name. "C" is set to either C1 or C2 and then the program "gosubs" to Line 3000.

"SC(I)" is array containing all sightseeing costs.

"S(I)" is array containing sightseeing miles.

"M(I)" is array containing miles traveled daily, not including sightseeing miles.

PROGRAM LISTING

```
1 CLEAR 2000:RESTORE:DIM SC(30):DIM D(35,35):DIM S(30):DIM M
(30)
10 REM *** TRIP PLANNING PROGRAM
20 REM *** BY SAM NEWHOUSE
30 REM *** COPYRIGHT 1978 BY SAM NEWHOUSE
40 REM *** ALTAIR BASIC VERSION 3.4
41 WIDTH30:PRINTCHR$(27);CHR$(54);CHR$(16);CHR$(22);"***  TR
IP PLANNER  ***":PRINT"THIS PROGRAM WILL HELP YOU PLAN YOUR
ITINERARY AND COMPUTE YOUR PROBABLE DRIVING COSTS.":PRINT"CO
NTAINED IN THE PROGRAM IS DATA DEVELOPED BY THE AAA ON AVERA
GE COSTS ";
42 PRINT"OF CAR OWNERSHIP, OPERATION, AND MAINTENANCE.":PRIN
T"THE PROGRAM ALSO CONTAINS A DISTANCE TABLE WHICH INCLUDES
THE NUMBER OF MILES BETWEEN 35 CITIES.":PRINT"USING THIS INF
ORMATION, YOU WILL BE ABLE TO ESTIMATE HOW MANY DAYS YOU WIL
L HAVE TO DRIVE";
43 PRINT" TO COVER THE TRIP YOU WANT TO TAKE.":INPUT A$:PRIN
TCHR$(27);CHR$(51);CHR$(16);CHR$(22);:PRINT:WIDTH60:NULL0:PR
INT
50 REM *** INITIALIZE
60 REM ***  MH IS MILES PER HOUR (AVERAGE)
70 MH=55:INPUT"MILES PER HOUR (AVERAGE) ";MH
72 REM *** MG IS HIGHWAY MILES PER GALLON
74 MG=27:INPUT"HIGHWAY MILES PER GALLON ";MG
80 REM *** PP IS NUMBER OF PEOPLE ON TRIP
82 PP=2:INPUT"NUMBER OF PEOPLE ON TRIP ";PP
84 REM *** PG IS PRICE PER GALLON
86 PG=.6:INPUT"PRICE PER GALLON ";PG
88 REM *** DT IS TOTAL DAYS ON TRIP
90 DT=14:INPUT"NUMBER OF DAYS IN TRIP ";DT
92 REM *** RM IS REMAINING MILES
94 RM=0
96 REM *** DC IS DAY COUNTER
98 DC=1
110 FOR I=1 TO 35
120 FOR J=1 TO 35
130 READ D(I,J)
140 NEXT J
150 NEXT I
160 C2=25:INPUT"STARTING CITY # (USE '0' TO OBTAIN LIST)";C2
165 C=C2:GOSUB 3000:PRINT:INPUT"CORRECT-";YN$
170 IF YN$="YES" THEN 200
175 IF YN$="NO" THEN 160
180 PRINT"USE YES OR NO.":GOTO 160
200 REM *** START OF DAILY LOOP
210 PRINT:PRINT:PRINTCHR$(16);CHR$(22);
220 PRINT"DAY #";DC;" OF ";DT
230 INPUT"HOW MANY HOURS OF HIGHWAY DRIVING TODAY";H:D1=H*MH
240 IF H=0 THEN 260
250 IF RM=0 THEN 260
255 PRINT"YOU HAVE ";RM;" MILES TO GO TO GET TO ";
256 C=C2:GOSUB 3000:RM=RM-(H*MH):PRINT:IF RM<=0 THEN PRINT"Y
OU HAVE ARRIVED IN  ";:C=C2:GOSUB 3000:PRINT
257 IFRM>0 THEN H=0:GOTO260
258 IF RM<0 THEN RM=RM*(-1):H=RM/MH:RM=0:GOTO260
```

Variable Costs	**Average per mile**
Gasoline and oil	4.11 cents
Maintenance	1.03 cents
Tires	0.66 cents
	5.80 cents

Fixed Costs	**Annually**
Comprehensive insurance ($50 ded.) . . . . . $	80.00
$100 ded. collision insurance	188.00
Property damage and liability ($100/300/25M) . .	250.00
License, registration, taxes	74.00
Depreciation	847.00
	$1,439.00
	(or $3.94 per day)

(For air conditioning, add .20¢ per mile and .20¢ per day.)

If financing is involved, add a daily amount to match the value of the interest over the life of the loan. For example, if you trade in your car, you might borrow $3000 at 10.5% rate of interest for 48 months. Your total interest payment would be $687.36, or 47¢ per day. (Note: Neither air conditioning nor financing is taken into account by the Planner program.)

```
259 IF RM=0 THEN H=0
260 IF H>0 THEN PRINT"YOU HAVE ";H;" HOURS OF HIGHWAY DRIVIN
G TIME LEFT IN DAY ";DC
265 IF RM=0 AND H>0 THEN GOSUB 2000:RM=RM-(H*MH):H=0:IF RM<0
 THEN D1=D1-ABS(RM):PRINT"YOU HAVE ARRIVED IN  ";:C=C2:GOSUB
 3000:PRINT::RM=0
266 PRINT"YOU DROVE ";D1; " MILES TODAY."
267 IF RM>0 THEN PRINT"NOW YOU HAVE ";RM;" MILES TO GO TO ";
:C=C2:GOSUB 3000:PRINT
268 INPUT A$
270 PRINT:PRINT:PRINTCHR$(16);CHR$(22);
275 ST=0:INPUT"HOW MANY MILES OF SIGHTSEEING CAR TRAVEL TODA
Y";ST
278 SC=0:INPUT"SIGHTSEEING COST ";SC
280 SC(DC)=SC:S(DC)=ST:M(DC)=D1:D1=D1+ST
285 PRINT:PRINT:PRINTCHR$(16);CHR$(22);
290 PRINT"COSTS FOR DAY ";DC
292 PRINT"MILES-";TAB(15);D1
295 PRINT"GAS + OIL";TAB(15);D1*.0411
300 PRINT"MAINTENANCE";TAB(15);D1*.0103
305 PRINT"TIRES";TAB(15);D1*.0066
310 PRINT"TOTAL-";TAB(15);D1*.058
312 PRINT
315  PRINT"FIXED COSTS-"
320 PRINT"INSURANCE,TAXES,DEPRECIATION-"
325 PRINT"   PER DAY-";TAB(15);3.94
330 IF PP>2 THEN RC=54 ELSE RC=27
340 PRINT"MOTEL LODGING-";TAB(15);RC
350 PRINT"FOOD COSTS-";TAB(15);PP*13
355 PRINT"SIGHTSEEING-";TAB(15);SC
360 PRINT"----------------------------"
370 PRINT"GRAND TOTAL-";TAB(15);(D1*.058)+3.94+RC+(PP*13)+SC
380 INPUT A$:DC=DC+1:IF DC>DT THEN 400
390 GOTO 200
400 REM *** FINAL TOTALS ROUTINE
410 PRINT:PRINT:PRINTCHR$(16);CHR$(22);
420  SC=0:ST=0:D1=0
430 FORI=1TODT
440 SC=SC+SC(I):ST=ST+S(I):D1=D1+M(I)
450 NEXT I
455 D2=D1+ST
460 FI=3.94*DT
470 RC=RC*DT
480 FC=(PP*13)*DT
490 TI=D2*.0066
500 MA=D2*.0103
510 GO=D2*.0411
520 TC=D2*.058
530 IF RM>0 THEN PRINT"YOU STILL HAVE ";RM;" MILES TO GO TO
GET TO ";:C=C2:GOSUB 3000:PRINT
540 PRINT"COSTS FOR YOUR ";DT;" DAY TRIP"
550 PRINT"HIGHWAY MILES DRIVEN-";D1
560 PRINT"SIGHTSEEING MILES DRIVEN-";ST
570 PRINT"TOTAL MILES DRIVEN-";D2
580 PRINT:PRINT"GAS & OIL";TAB(15);GO
590 PRINT"MAINTENANCE";TAB(15);MA
600 PRINT"TIRES";TAB(15);TI
610 PRINT"----------------------------"
620 PRINT"TOTAL-";TAB(15);D2*.058
630 PRINT:PRINT"FIXED COSTS-"
640 PRINT"INSURANCE, TAXES, DEPRECIATION-";FI
650 PRINT"MOTEL LODGING-";RC
660 PRINT"FOOD COSTS-";FC
670 PRINT"SIGHTSEEING-";SC
680 PRINT"----------------------------"
690 T=SC+FC+RC+FI+(D2*.058)
700 PRINT"GRAND TOTAL";TAB(15);T
710 END
2000 REM *** INPUT CITIES ROUTINE
2010 PRINT"STARTING AT ";
2020 C=C2:GOSUB 3000
2025 PRINT:C1=C2
2030 INPUT"ENDING AT CITY #";C2
2040 C=C2:GOSUB 3000
2045 YN$="*":INPUT"--CORRECT ";YN$:IF YN$="YES" THEN 2050
2047 IF YN$="NO" THEN 2030
2048 PRINT"USE YES OR NO.":GOTO2045
2050 D=D(C1,C2)
2055 PRINT CHR$(16);CHR$(22);
2057 PRINT"FROM ";:C=C1:GOSUB3000:PRINT" -TO- ";:C=C2:GOSUB
3000:PRINT"."
2060 PRINT"DISTANCE=";D;" MILES."
2065 RM=D
2070  PRINT"AT AN AVERAGE OF ";MH;" M.P.H."
2080 PRINT"THIS WILL TAKE ";D/MH;" HOURS."
2130 RETURN
3000 REM *** PRINT CITY NAME
3002 IF C=0 THEN 4000
3005 IF C>35 THEN PRINT"BAD CITY NUMBER":RETURN
3010 ON C GOSUB 3030,3035,3040,3045,3050,3055,3060,3065,3070
,3075,3080,3085,3090,3095,3100,3105,3110,3115,3120,3125,3130
,3135,3140,3145,3150,3155,3160,3165,3170,3175,3180,3185,3190
,3195,3200
3020 RETURN
3030 PRINT"ATLANTA";:RETURN
3035 PRINT"BALTIMORE";:RETURN
3040 PRINT"BIRMINGHAM";:RETURN
3045 PRINT"BOSTON";:RETURN
3050 PRINT"BUFFALO";:RETURN
3055 PRINT"CHICAGO";:RETURN
3060 PRINT"CINCINNATI";:RETURN
3065 PRINT"CLEVELAND";:RETURN
3070 PRINT"DALLAS";:RETURN
3075 PRINT"DENVER";:RETURN
3080 PRINT"DETROIT";:RETURN
3085 PRINT"HARRISBURG";:RETURN
3090 PRINT"HOUSTON";:RETURN
3095 PRINT"INDIANAPOLIS";:RETURN
3100 PRINT"KANSAS CITY";:RETURN
3105 PRINT"LAS VEGAS";:RETURN
3110 PRINT"LOS ANGELES";:RETURN
3115 PRINT"LOUISVILLE";:RETURN
3120 PRINT"MEMPHIS";:RETURN
3125 PRINT"MEXICO CITY";:RETURN
3130 PRINT"MIAMI";:RETURN
3135 PRINT"MONTREAL";:RETURN
```

(Continued on following page)

For mileage in excess of 15,000 annually, an additional depreciation allowance of $36 per thousand should be added to the fixed costs.

10,000 miles @5.80¢	$ 580.00
365 days @$3.94	1,439.00
	$2,019.00

(or 20.2¢ per mile)

In contrast, a car driven 20,000 miles a year would cost:

20,000 miles @5.80¢	$1,160.00
365 days @$3.94	1,439.00
5,000 miles @$36/thousand	180.00
	$2,779.00

(or 13.9¢ per mile)

With gasoline prices increasing, you should remember that for every 10¢ per gallon increase, the per mile cost of running a car increases by one cent (if your car delivers 10 miles per gallon). If your car delivers 20 miles per gallon, the cost increase per mile is one-half cent.

If your car differs substantially from the car described by the program, adjust the figures accordingly.

Listing continued

```
3140 PRINT"NASHVILLE";:RETURN
3145 PRINT"NEW ORLEANS";:RETURN
3150 PRINT"NEW YORK CITY";:RETURN
3155 PRINT"OMAHA";:RETURN
3160 PRINT"PHILADELPHIA";:RETURN
3165 PRINT"PHOENIX";:RETURN
3170 PRINT"PORTLAND";:RETURN
3175 PRINT"QUEBEC";:RETURN
3180 PRINT"ST. LOUIS";:RETURN
3185 PRINT"SALT LAKE CITY";:RETURN
3190 PRINT"SAN FRANCISCO";:RETURN
3195 PRINT"SEATTLE";:RETURN
3200 PRINT"WASHINGTON D.C.";:RETURN
4000 PRINT"LIST OF CITIES"
4005 PRINT:PRINT"CITY";TAB(20);"NUMBER"
4010 FOR C=1 TO 35
4015 GOSUB 3000:PRINT TAB(20);C
4020 NEXT C
4030 PRINT"END OF LIST OF CITIES"
4040 RETURN
9000 REM *** 35 CITIES DISTANCE TABLE
9010 REM *** ATLANTA
9020 DATA 0,645,150,1037,859,674,440,672,795,1398,699,700,78
9,493,798,1964,2182,382,371,1768,655,1181,242,479,841,986,74
1,1793,2601,1331,541,1878,2496,2618,608
9030 REM *** BALTIMORE (2)
9040 DATA 645,0,773,392,346,668,497,343,1356,1621,503,75,141
2,563,1048,2398,2636,598,904,2391,1112,542,696,1115,196,1113
,96,2261,2751,686,798,2044,2796,2681,37
9050 REM *** BIRMINGHAM (3)
9060 DATA 150,773,0,1165,896,642,465,709,645,1286,724,800,63
9,475,699,1834,2032,364,246,1618,751,1276,196,342,969,898,86
9,1643,2505,1431,465,1781,2366,2535,736
9070 REM *** BOSTON (4)
9080 DATA 1037,392,1165,0,446,963,840,628,1748,1949,695,373,
1804,906,1391,2725,2779,941,1296,2783,1504,318,1088,1507,206
,1412,296,2604,3046,384,1141,2343,3095,2976,429
9090 REM *** BUFFALO (5)
9100 DATA 859,346,896,446,0,522,431,187,1346,1508,253,278,14
60,481,966,2284,2554,532,899,2438,1409,377,700,1217,372,971,
353,2179,2605,538,716,1902,2654,2535,356
9110 REM *** CHICAGO (6)
9120 DATA 674,668,642,963,522,0,287,335,917,996,266,639,1067
,181,499,1772,2054,292,530,2045,1329,828,446,912,802,459,738
,1713,2083,989,289,1390,2142,2013,671
9130 REM *** CINCINNATI (7)
9140 DATA 440,497,465,840,431,287,0,244,920,1164,259,468,102
9,106,591,1941,2179,101,468,2007,1095,805,269,786,647,693,56
7,1804,2333,966,340,1610,2362,2300,481
9150 REM *** CLEVELAND (8)
9160 DATA 692,343,709,628,187,335,244,0,1159,1321,170,314,12
73,294,779,2097,2367,345,712,2251,1264,561,513,1030,473,784,
413,1992,2418,722,529,1715,2467,2348,346
9170 REM *** DALLAS (9)
9180 DATA 795,1356,645,1748,1346,917,920,1159,0,781,1143,138
3,243,865,489,1221,1387,819,452,1138,1300,1705,660,496,1552,
644,1452,998,2009,1866,630,1242,1753,2078,1319
9190 REM *** DENVER (10)
9200 DATA 1398,1621,1286,1949,1508,996,1164,1321,781,0,1253,
1592,1019,1058,600,777,1059,1120,1040,1746,2037,1815,1156,12
73,1771,537,1691,792,1238,1976,857,504,1235,1307,1616
9210 REM *** DETROIT (11)
9220 DATA 699,503,724,695,253,266,259,170,1143,1253,0,474,12
65,278,743,2029,2311,360,713,2243,1352,562,528,1045,637,716,
573,1957,2349,723,513,1647,2399,2279,506
9230 REM HARRISBURG (12)
9240 DATA 700,75,800,373,278,639,468,314,1383,1592,474,0,143
9,534,1019,2369,2607,569,931,2418,1182,490,723,1142,180,1084
,102,2232,2722,634,769,2015,2767,2652,107
9250 REM *** HOUSTON (13)
9260 DATA 789,1412,639,1804,1460,1067,1029,1273,243,1019,126
5,1439,0,987,710,1417,1538,928,561,979,1190,1827,769,356,160
8,865,1508,1149,2205,1988,779,1438,1912,2274,1375
9270 REM *** INDIANAPOLIS (14)
9280 DATA 493,563,475,906,481,181,106,294,865,1058,278,534,9
87,0,485,1835,2073,111,435,1965,1148,840,279,796,713,587,633
,1698,2227,1001,235,1504,2256,2194,558
9290 REM *** KANSAS CITY, MISSOURI (15)
9300 DATA 798,1048,697,1391,966,499,591,779,489,600,743,1019
,710,485,0,1365,1589,520,451,1627,1448,1305,556,806,1198,201
,1118,1214,1809,1466,257,1086,1835,1839,1043
9310 REM *** LAS VEGAS (16)
9320 DATA 1964,2398,1834,2725,2284,1772,1941,2097,1221,777,2
029,2369,1417,1835,1365,0,282,1884,1593,1834,2521,2591,1801,
1717,2548,1313,2468,285,981,2752,1621,433,564,1152,2393
9330 REM *** LOS ANGELES (17)
9340 DATA 2182,2636,2032,2779,2554,2054,2179,2367,1387,1059,
2311,2607,1538,2073,1589,282,0,2108,1817,1917,2687,2873,2025
,1883,2786,1595,2706,389,959,3034,1845,715,379,1131,2631
9350 REM *** LOUISVILLE (18)
9360 DATA 382,598,364,941,532,292,101,345,819,1120,360,569,9
28,111,520,1884,2108,0,367,1906,1037,906,168,685,748,687,668
,1733,2320,1067,263,1597,2349,2305,582
9370 REM *** MEMPHIS (19)
9380 DATA 371,904,246,1296,899,530,468,712,452,1040,713,931,
561,435,451,1593,1817,367,0,1539,997,1273,208,390,1100,652,1
000,1442,2259,1434,285,1535,2125,2290,867
9390 REM *** MEXICO CITY (20)
9400 DATA 1768,2391,1618,2783,2438,2045,2007,2251,1138,1746,
2243,2918,979,1965,1627,1834,1917,1906,1539,0,2169,2805,1747
,1335,2587,1782,2487,1549,2783,2966,1757,2016,2291,2852,2354
9410 REM *** MIAMI (21)
9420 DATA 655,1112,751,1504,1409,1329,1095,1264,1300,2037,13
52,1182,1190,1148,1448,2521,2687,1037,997,2169,0,1654,897,85
6,1308,1641,1208,2298,3256,1798,1196,2532,3053,3273,1075
9430 REM *** MONTREAL (22)
9440 DATA 1181,542,1276,318,377,828,805,561,1705,1815,562,49
0,1827,840,1305,2591,2873,906,1273,2805,1654,0,1074,1591,378
,1278,449,2519,2755,161,1075,2209,2961,2685,579
9450 REM *** NASHVILLE (23)
9460 DATA 242,696,196,1088,700,446,269,513,660,1156,528,723,
796,279,556,1801,2025,168,208,1747,897,1074,0,517,892,744,79
2,1650,2359,1235,299,1636,2333,2376,659
9470 REM *** NEW ORLEANS (24)
9480 DATA 479,1115,342,1507,1217,912,786,1030,496,1273,1045,
1142,356,796,806,1717,1883,685,390,1335,856,1591,517,0,1311,
1007,1211,1494,2505,1752,673,1738,2249,2574,107
9490 REM *** NEW YORK CITY (25)
```

Using Planner

You initialize the program by responding to such questions as: your average mph, number of people on the trip, number of days in trip and your starting city. At any time, a list of the 35 cities incorporated into the program can be obtained by inputting '0' as the city #.

For each day of your trip you must first decide how many (highway) hours you want to drive. Input that information, then respond to the question, "where you want to go?"

After selecting and inputting your destination, the program will tell you the total distance and number of hours required to make the trip. The program then subtracts the number of miles you traveled that day from the total distance, and tells you how far you still have to go, if any.

Finally, the computer will print a breakdown of your mileage and expenses for that day.

The next "day" you begin again by inputting the number of hours you want to drive for that day. Then, the program checks to see if you have any distance remaining to reach your destination. If you still have miles left to travel, the program will allow you to go as far toward your destination as your driving limit per day will allow. If you reach your destination this

```
9500 DATA 841,196,969,206,372,802,647,473,1552,1771,637,180,
1608,713,1198,2548,2786,748,1100,2587,1308,378,892,1311,0,12
51,100,2411,2885,518,948,2182,2934,2815,233
9510 REM *** OMAHA (26)
9520 DATA 986,1113,898,1412,971,459,693,784,644,537,716,1084
,865,587,201,1313,1595,687,652,1782,1641,1278,744,1007,1251,
0,1183,1290,1654,1439,449,931,1683,1638,1116
9530 REM *** PHILADELPHIA (27)
9540 DATA 741,96,869,296,353,738,567,413,1452,1691,573,102,1
508,633,1118,2468,2706,668,1000,2487,1208,449,792,1211,100,1
183,0,2331,2821,593,868,2114,2866,2751,133
9550 REM *** PHOENIX (28)
9560 DATA 1793,2261,1643,2604,2179,1713,1804,1992,998,792,19
57,2232,1149,1698,1214,285,389,1733,1442,1549,2298,2519,1650
,1494,2411,1290,2331,0,1266,2680,1470,648,763,1437,2256
9570 REM *** PORTLAND (29)
9580 DATA 2601,2751,2505,3046,2605,2083,2333,2418,2009,1238,
2349,2722,2205,2227,1809,981,959,2320,2259,2783,3256,2755,23
59,2505,2885,1654,2821,1266,0,2903,2060,767,636,172,2754
9590 REM *** QUEBEC (30)
9600 DATA 1331,686,1431,384,538,989,966,722,1866,1976,723,63
4,1988,1001,1466,2752,3034,1067,1434,2966,1798,161,1235,1752
,518,1439,593,2680,2903,0,1236,2370,3122,2833,723
9610 REM *** ST. LOUIS (31)
9620 DATA 541,798,465,1141,716,289,340,529,630,857,1513,769,
779,235,257,1621,1845,263,285,1757,1196,1075,299,673,948,449
,868,1470,2060,1236,0,1337,2089,2081,793
9630 REM *** SALT LAKE CITY (32)
9640 DATA 1878,2044,1781,2343,1902,1390,1610,1715,1242,504,1
647,2015,1438,1504,1086,433,715,1597,1535,2016,2532,2209,163
6,1738,2182,931,2114,648,767,2370,1337,0,752,836,2047
9650 REM *** SAN FRANCISCO (33)
9660 DATA 2496,2796,2366,3095,2654,2142,2362,2367,1753,1235,
2399,2767,1912,2256,1835,564,379,2349,2125,2291,3053,2961,23
33,2249,2934,1683,2866,763,636,3122,2089,752,0,808,2799
9670 REM *** SEATTLE (34)
9680 DATA 2618,2681,2535,2976,2535,2013,2300,2348,2078,1307,
2279,2652,2274,2194,1839,1152,1131,2305,2290,2852,3273,2685,
2376,2574,2815,1638,2751,1437,172,2833,2081,836,808,0,2684
9690 REM *** WASHINGTON D.C. (35)
9700 DATA 608,37,736,429,356,671,481,346,1319,1616,506,107,1
375,558,1043,2393,2631,582,867,2354,1075,579,659,1078,233,11
16,113,2256,2754,723,793,2047,2799,2684,0
```

PROGRAM RUN

```
***   TRIP PLANNER   ***
THIS PROGRAM WILL HELP YOU PLA
N YOUR ITINERARY AND COMPUTE Y
OUR PROBABLE DRIVING COSTS.
CONTAINED IN THE PROGRAM IS DA
TA DEVELOPED BY THE AAA ON AVE
RAGE COSTS OF CAR OWNERSHIP, O
PERATION, AND MAINTENANCE.
THE PROGRAM ALSO CONTAINS A DI
STANCE TABLE WHICH INCLUDES TH
E NUMBER OF MILES BETWEEN 35 C
ITIES.
USING THIS INFORMATION, YOU WI
LL BE ABLE TO ESTIMATE HOW MAN
Y DAYS YOU WILL HAVE TO DRIVE
TO COVER THE TRIP YOU WANT TO
TAKE.
?

MILES PER HOUR (AVERAGE) ? 5500
HIGHWAY MILES PER GALLON ? 3300
NUMBER OF PEOPLE ON TRIP ? 22
PRICE PER GALLON ? ..6600
NUMBER OF DAYS IN TRIP ? 77
STARTING CITY # (USE '0' TO OBTAIN LIST)? 00
LIST OF CITIES

CITY              NUMBER
ATLANTA           1
BALTIMORE         2
BIRMINGHAM        3
BOSTON            4
BUFFALO           5
CHICAGO           6
CINCINNATI        7
CLEVELAND         8
DALLAS            9
DENVER            10
DETROIT           11
HARRISBURG        12
HOUSTON           13
INDIANAPOLIS      14
KANSAS CITY       15
LAS VEGAS         16
LOS ANGELES       17
LOUISVILLE        18
MEMPHIS           19
MEXICO CITY       20
MIAMI             21
MONTREAL          22
NASHVILLE         23
NEW ORLEANS       24
NEW YORK CITY     25
OMAHA             26
PHILADELPHIA      27
PHOENIX           28
PORTLAND          29
QUEBEC            30
ST. LOUIS         31
SALT LAKE CITY    32
SAN FRANCISCO     33
SEATTLE           34
WASHINGTON D.C.   35
END OF LIST OF CITIES

CORRECT-? NNOO
STARTING CITY # (USE '0' TO OBTAIN LIST)? 2255
NEW YORK CITY
CORRECT-? YYEESS
```

day, and there is still driving time left, you may enter a new destination for the remainder of the day.

However, if you have driving hours left over, and you do not want to go anywhere else this day, enter the same city as starting point and destination point.

This continues until you have traveled the total number of days of your trip.

At this point, your total mileage and cost figures are printed out, and the program ends.

Some possible extensions to the program, left as an exercise for the reader, include calculating extra costs for air conditioning, calculating car financing costs per day and adding latitude and longitude figures for each city (this would allow the program to calculate your approximate location at the end of each day, and compare it with a stored list of locations of points of interest).

You might also add extra cities to the distance table (this can be done by extracting information from a road atlas. For each city, its data statement would contain the distance from that city to all the other cities on your list. Change lines 110-120, lines 3000-3200 and line 4010 to reflect these changes.

You can also calculate toll costs by using information found in a road atlas. Each state has differing costs per mile of highway. PC

DAY 1 OF 7

```
DAY # 1  OF  7
HOW MANY HOURS OF HIGHWAY DRIVING TODAY? 1100
YOU HAVE  10  HOURS OF HIGHWAY DRIVING TIME LEFT IN DAY  1
STARTING AT NEW YORK CITY
ENDING AT CITY #? 1155
KANSAS CITY--CORRECT ? YYEESS
FROM NEW YORK CITY -TO- KANSAS CITY.
DISTANCE= 1198  MILES.
AT AN AVERAGE OF  50  M.P.H.
THIS WILL TAKE  23.96  HOURS.
YOU DROVE  500  MILES TODAY.
NOW YOU HAVE  698  MILES TO GO TO KANSAS CITY
?

HOW MANY MILES OF SIGHTSEEING CAR TRAVEL TODAY? 00
SIGHTSEEING COST ? 00

COSTS FOR DAY  1
MILES-          500
GAS + OIL       20.55
MAINTENANCE     5.15
TIRES           3.3
TOTAL-          29

FIXED COSTS-
INSURANCE,TAXES,DEPRECIATION-
   PER DAY-     3.94
MOTEL LODGING-  27
FOOD COSTS-     26
SIGHTSEEING-    0
-----------------------------
GRAND TOTAL-    85.94
?
```

DAY 2 OF 7

```
DAY # 2  OF  7
HOW MANY HOURS OF HIGHWAY DRIVING TODAY? 1122
YOU HAVE  698  MILES TO GO TO GET TO KANSAS CITY
YOU DROVE  600  MILES TODAY.
NOW YOU HAVE  98  MILES TO GO TO KANSAS CITY
?

HOW MANY MILES OF SIGHTSEEING CAR TRAVEL TODAY? 110000
SIGHTSEEING COST ? 5500

COSTS FOR DAY  2
MILES-          700
GAS + OIL       28.77
MAINTENANCE     7.21
TIRES           4.62
TOTAL-          40.6

FIXED COSTS-
INSURANCE,TAXES,DEPRECIATION-
   PER DAY-     3.94
MOTEL LODGING-  27
FOOD COSTS-     26
SIGHTSEEING-    50
----------------------------
GRAND TOTAL-    147.54
?
```

DAY 3 OF 7

```
DAY # 3  OF  7
HOW MANY HOURS OF HIGHWAY DRIVING TODAY? 1100
YOU HAVE  98  MILES TO GO TO GET TO KANSAS CITY
YOU HAVE ARRIVED IN  KANSAS CITY
YOU HAVE  8.04  HOURS OF HIGHWAY DRIVING TIME LEFT IN DAY
 3
STARTING AT KANSAS CITY
ENDING AT CITY #? 1188
LOUISVILLE--CORRECT ? YYEESS
FROM KANSAS CITY -TO- LOUISVILLE.
DISTANCE= 520  MILES.
AT AN AVERAGE OF  50  M.P.H.
THIS WILL TAKE  10.4  HOURS.
YOU DROVE  500  MILES TODAY.
NOW YOU HAVE  118  MILES TO GO TO LOUISVILLE
?


HOW MANY MILES OF SIGHTSEEING CAR TRAVEL TODAY? 5500
SIGHTSEEING COST ? 1100


COSTS FOR DAY  3
MILES-          550
GAS + OIL       22.605
MAINTENANCE     5.665
TIRES           3.63
TOTAL-          31.9

FIXED COSTS-
INSURANCE,TAXES,DEPRECIATION-
   PER DAY-     3.94
MOTEL LODGING-  27
FOOD COSTS-     26
SIGHTSEEING-    10
----------------------------
GRAND TOTAL-    98.84
?
```

DAY 4 OF 7

```
DAY # 4  OF  7
HOW MANY HOURS OF HIGHWAY DRIVING TODAY? 88
YOU HAVE  118  MILES TO GO TO GET TO LOUISVILLE
YOU HAVE ARRIVED IN  LOUISVILLE
YOU HAVE  5.64  HOURS OF HIGHWAY DRIVING TIME LEFT IN DAY
 4
STARTING AT LOUISVILLE
ENDING AT CITY #? 22
```

```
BALTIMORE--CORRECT ? YYEESS
FROM LOUISVILLE -TO- BALTIMORE.
DISTANCE= 598  MILES.
AT AN AVERAGE OF  50  M.P.H.
THIS WILL TAKE  11.96  HOURS.
YOU DROVE  400  MILES TODAY.
NOW YOU HAVE  316  MILES TO GO TO BALTIMORE
?


HOW MANY MILES OF SIGHTSEEING CAR TRAVEL TODAY? 00
SIGHTSEEING COST ? 00


COSTS FOR DAY  4
MILES-          400
GAS + OIL       16.44
MAINTENANCE     4.12
TIRES           2.64
TOTAL-          23.2

FIXED COSTS-
INSURANCE,TAXES,DEPRECIATION-
   PER DAY-     3.94
MOTEL LODGING-  27
FOOD COSTS-     26
SIGHTSEEING-    0
-----------------------------
GRAND TOTAL-    80.14
?
```

DAY 5 OF 7

```
DAY # 5  OF  7
HOW MANY HOURS OF HIGHWAY DRIVING TODAY? 1100
YOU HAVE  316  MILES TO GO TO GET TO BALTIMORE
YOU HAVE ARRIVED IN  BALTIMORE
YOU HAVE  3.68  HOURS OF HIGHWAY DRIVING TIME LEFT IN DAY
 5
STARTING AT BALTIMORE
ENDING AT CITY #? 1122
HARRISBURG--CORRECT ? YYEESS
FROM BALTIMORE -TO- HARRISBURG.
DISTANCE= 75  MILES.
AT AN AVERAGE OF  50  M.P.H.
THIS WILL TAKE  1.5  HOURS.
YOU HAVE ARRIVED IN  HARRISBURG
YOU DROVE  391  MILES TODAY.
?


HOW MANY MILES OF SIGHTSEEING CAR TRAVEL TODAY? 110000
SIGHTSEEING COST ? 1155


COSTS FOR DAY  5
MILES-          491
GAS + OIL       20.1801
MAINTENANCE     5.0573
TIRES           3.2406
TOTAL-          28.478

FIXED COSTS-
INSURANCE,TAXES,DEPRECIATION-
   PER DAY-     3.94
MOTEL LODGING-  27
FOOD COSTS-     26
SIGHTSEEING-    15
-----------------------------
GRAND TOTAL-    100.418
?
```

DAY 6 OF 7

```
DAY # 6  OF  7
HOW MANY HOURS OF HIGHWAY DRIVING TODAY? 00
YOU DROVE  0  MILES TODAY.
?


HOW MANY MILES OF SIGHTSEEING CAR TRAVEL TODAY? 110000
SIGHTSEEING COST ? 2255

COSTS FOR DAY  6
MILES-          100
GAS + OIL       4.11
MAINTENANCE     1.03
TIRES           .66
TOTAL-          5.8

FIXED COSTS-
INSURANCE,TAXES,DEPRECIATION-
   PER DAY-     3.94
MOTEL LODGING-  27
FOOD COSTS-     26
SIGHTSEEING-    25
-----------------------------
GRAND TOTAL-    87.74
?
```

DAY 7 OF 7

```
DAY # 7  OF  7
HOW MANY HOURS OF HIGHWAY DRIVING TODAY? 22
YOU HAVE  2  HOURS OF HIGHWAY DRIVING TIME LEFT IN DAY  7
STARTING AT HARRISBURG
ENDING AT CITY #? 2255
NEW YORK CITY--CORRECT ? YYEESS
FROM HARRISBURG -TO- NEW YORK CITY.
DISTANCE= 180  MILES.
AT AN AVERAGE OF  50  M.P.H.
THIS WILL TAKE  3.6  HOURS.
YOU DROVE  100  MILES TODAY.
NOW YOU HAVE  80  MILES TO GO TO NEW YORK CITY
?


HOW MANY MILES OF SIGHTSEEING CAR TRAVEL TODAY? 00
SIGHTSEEING COST ? 00


COSTS FOR DAY  7
MILES-          100
GAS + OIL       4.11
MAINTENANCE     1.03
TIRES           .66
TOTAL-          5.8

FIXED COSTS-
INSURANCE,TAXES,DEPRECIATION-
   PER DAY-     3.94
MOTEL LODGING-  27
FOOD COSTS-     26
SIGHTSEEING-    0
-----------------------------
GRAND TOTAL-    62.74
?


YOU STILL HAVE  80  MILES TO GO TO GET TO NEW YORK CITY
COSTS FOR YOUR  7  DAY TRIP
HIGHWAY MILES DRIVEN- 2491
SIGHTSEEING MILES DRIVEN- 350
TOTAL MILES DRIVEN- 2841

GAS & OIL       116.765
MAINTENANCE     29.2623
TIRES           18.7506
-----------------------------
TOTAL-          164.778

FIXED COSTS-
INSURANCE, TAXES, DEPRECIATION- 27.58
MOTEL LODGING- 189
FOOD COSTS- 182
SIGHTSEEING- 100
-----------------------------
GRAND TOTAL     663.358

OK
```

C.P.

BY Y. KNOT

Until recently, George had two intellectual hang-ups – intelligent life and minicomputers. Of the former he found little evidence, and about the latter he was just learning, when he decided to try the best of both. He would build an intelligent computer. It would be capable of learning on its own and conversing in the local dialect, rather than Fo-CotaBol, or whatever.

To this end he bought a kit computer and began, with a lot of TLC, to assemble it in a manner closely resembling the method used by Mother Nature. First there was the mating of just two resistors. Then, gradually, over a period of nine months, he built his "comfetus".

Knowing that human fetal brainwaves begin sometime close to sixteen weeks, George plugged in the memory bank at sixteen weeks. This was not easy because he had the entire bus wired and plugged into a wall socket at twelve weeks. Fetal hearts usually start at about twelve weeks.

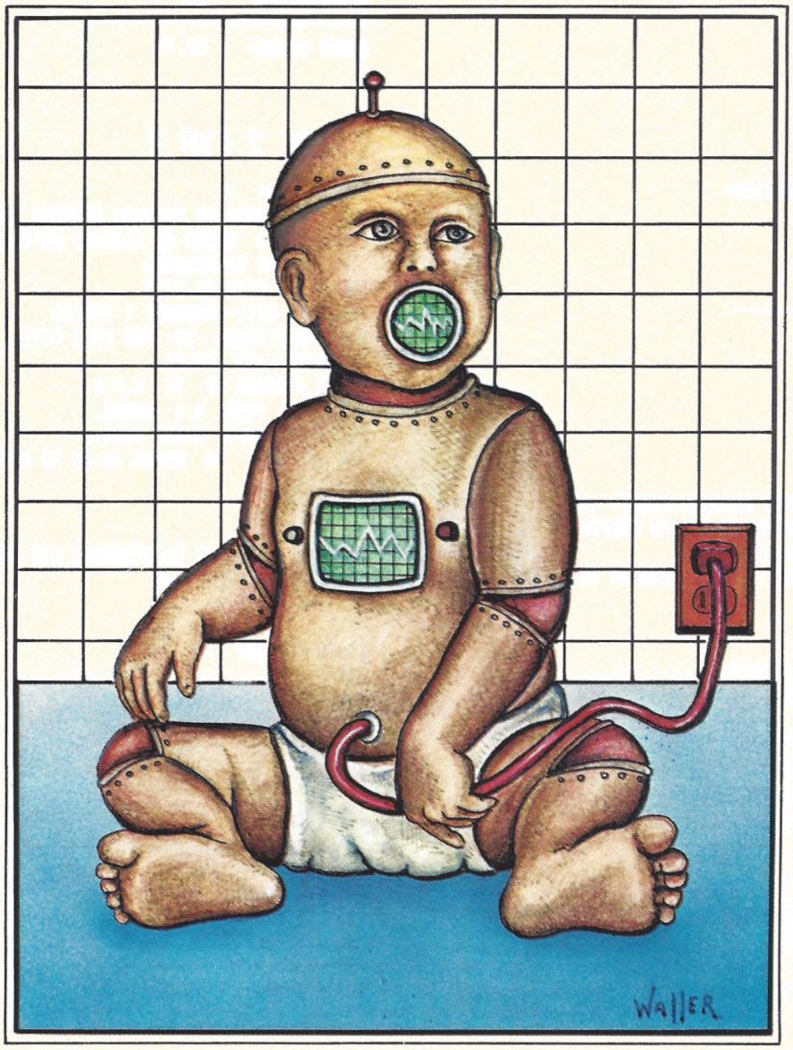


Illustration by Charles Waller

On D-Day (estimated delivery date, nine months and two weeks), George picked up his entire creation, slapped it twice on the CPU, and plugged in some rechargable batteries after cutting the umbilical cord to the wall socket. He also wired a black and white TV camera, microphone and speaker into his assembly so his baby could see

And communicate it did. The deafening wail lasted several minutes while George scrambled to separate and then balance the microphone and speaker. From then on he heard only occasional whimpers.

Every day, George talked to his comperson, in hopes of hearing it talk back. Three times a day, and once at 2 each morning, he recharged its batteries and cleaned off the dust which inevitably accumulated around the cooling fan.

Like a good father, George bought it toys, and read it books but the most it did was to say "hummmmmmmm," whenever he tickled it. The display screen occasionally curled the ends of its continuous saw-tooth pattern when he entered the room, but not much else happened for a long time.

Then one day George noticed his little creation rocking on its stand, and he immediately built it a carriage on wheels which it could control by itself. He spent weeks following it around, putting breakables on higher shelves, so they would not be smashed, and cleaning up the messes it made in closets, cabinets and every room of the house.

Eventually, his curious little comperson learned its way around and how to recharge itself.

Late one evening, George returned home from the computer store with some new games for his offspring, only to find it still and quiet in the middle of the family room, a ghost of an image on the display screen. He ran to it and checked its pulse; the batteries were at critically low levels. It was also running a temperature – the fan and cooling assembly had broken down.

With the skill of a pediatric surgeon, George quickly repaired the fan and cooled his little one, saving its life. Then he recharged the batteries. To his surprise, the comperson responded, "Thanks, Dad." Comperson, or "C.P." as he's known today, experienced logarithmic growth and learning from that time on.

Most of this story took place years ago, and now George has a whole family of compersons. His only problems seem to be keeping the oldest from borrowing the car too often, and the youngest from hanging out with a gang of uneducated Selectrics.

Wall Street Wallop

BY HERBERT SCHILDT

For a number of years, large firms and corporations have been doing market investment research of a statistical nature, making it increasingly difficult for the individual investor to compete successfully in the stock market. More than a few of us have had some rather bad, and at times shocking, experiences with the market. Let's face facts: the better informed the investor is the greater the likelihood of success.

Now, with the personal computer, it's possible to compete with, and even "beat", the corporate analysts at their own game! With the information and programs in this article, plus some good judgement, you should be able to increase your return on investments.

Statistical Techniques and procedures

To use statistical analysis successfully, you must have a grasp of certain fundamental principles. None of the operations or parameters are difficult. In fact they are, for the most part, intuitive. We will employ three quantities: arithmetic average, standard deviation and coefficient of correlation, and one major technique: regression line analysis.

The "arithmetic average" is known to most of us as "the average". The most common average, it's easy to compute by just summing the values used and dividing by the number of values.

Be careful not to confuse arithmetic average with mid-point or central value. And remember the arithmetic average is *not* necessarily the most common value. Example: both (40,40,40) and (120,0,0) have the same average: 40. But the sets of data are strikingly different.

Standard deviation is a less well-known quantity, but it's based on the concept of deviance from a central value. As stated above, the average can be misleading. Standard deviation tells us just how much the values differ from the average – and, indirectly, the probability of a certain value occuring. Standard deviation is computed thus: take the square root of the sum of the squared differences divided by the number of values. When written as an equation it looks like this:

$$S = \sqrt{\frac{1}{N-1}\sum (X-A)^2}$$

Here X represents each individual value, A the arithmetic average, N the number of measurements and S the standard deviation.

Standard deviation is important in two ways. First it tells us the spread of the data being sampled – just how different each value is from the average value. And it tells the probability of a value being within one, two or three deviations of the average, or a percentage thereof.

To explain standard deviation, use the "Law of Normality" and the properties of the "bell shaped" or normal curve. The "Law of Normality" states that events occurring in nature tend to describe a normal probability distribution – the "bell" curve. This curve has two interesting and useful qualities: the mid-point or "peak" is the average and there are definite probabilities associated with the directed distance a value is from the average.

Without laboring a lengthly explanation of why, the probabilities are: 68.3% that any given random value will lie within one standard deviation from the mean (average), 95.4% that is within two deviations, and 99.7% that it is within three (see Figure 1).

For example, let's assume a normal distribution with a mean of 10 and a standard deviation of 1. We can say that there's a 68.3% chance that any randomly selected value from that sample will be within one deviation from the mean, or in this case between 9 and 11, and 95.4% chance of being within two deviations, or between 8 and 12. Understand that if we want the probability of a value being one standard deviation greater than the mean, we have a 34.15% chance.

Once again, the odds are on a value being plus or minus a deviation from the mean. There are further refinements you can add to relate the deviation in a broader sense, but they lie outside the scope of this discussion.

The coefficient of correlation is the measure of the strength or closeness of the relation be-

MARKA Program Notes

The analysis program uses three separate data files on disk. The first is the Index file which can include any Index, such as the Dow, Amex, S & P 500, and so on. However, each must be discreet and each must end with the value -1. In order to simplify analysis, I organize all the Index files for the same period of time.

The second data file is the Corporation data files. These files hold the price per share, volume, P/E, and % yield of each individual issue. They must be kept separate and cannot be mixed together on a file. The data *must* (!!!) be in the following order: price : volume : P/E ratio : % yield. Any other order will cause errors. *All* file names must be 5 letters.

The last file is the Processed Parameters data file. This file must be initialized with the names of the corporations and zeros in the variables. In North Star BASIC the files are formated as follows: READ #2 %60*H5, S$, A, S, H1, L1, A1, S1, R, A2, A4.

S$ = name of corporation, A = Index average, S = standard deviation of Index, H1 = high price per share, L1 = low price per share, A1 = average price per share, S1 = standard deviation price per share R = correlation coefficient, A2 = volume average. These parameters must be in the above order or errors will occur.

Conversion to other disk based BASICs should prove no problem. Conversion to cassette tape system should be possible. Either put data into the program DATA lines (not suggested), or use the cassette data files, although execution time will suffer.

The program itself is straightforward and should provide little problem in converting to any BASIC. Although the graph requires direct cursor addressing, this feature could be modified. Also, if your terminal does not have a "clear screen" command, delete this command in line 1660 and use a return or other routine.

The program requires 24K of RAM. You might save some space by deleting the REM statements; however, to make it fit into 4K of memory, the program will have to be broken into pieces.

To repeat: corporate data file names *must* be 5 letters long, the Index file's data *must* end with -1 and the Processed Parameter file *must* be initialized before attempting to write to it.

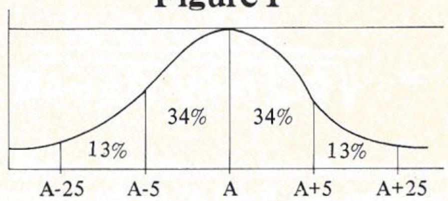
Marka Program Listing

```
10 LINE 80
20 Q1=1\Q2=1\Q3=1\Q4=1
30 M=0
40 T=0 \ T1=0 \ T2=0 \ T3=0 \ T4 =0
50 A=0 \A1=0 \A2=0 \ A3=0\ A4=0
60 W=0 \W1=0 \ W2=0 \ W3=0 \ W4=0
70 V=0 \ V1=0 \ V2=0\ V3=0\  V4=0
80 S=0 \ S1=0 \S2=0 \S3=0 \S4=0
90 Q5=1\ Q6=1 \Q7=1\Q8=1
100 Y=0 \X=0 \R=0 \ B=0 \ A5=0
110 N=0
120 Q9=1\E9=1
130 IF M1=1 THEN 380
140 REM THIS IS THE MARKET ANALYSIS PROGRAM
150 GOSUB 1660\ REM THE CLEAR SCREEN SUBROUTINE
160 !TAB(25), "MARKET ANALYSIS PROGRAM"
170 !
180 !
190 !TAB(10),"THIS PROGRAM WILL ANALIZE DATA AND YEILD THE FOLLOWING: "
200 !
210 !
220 !"I.           HIGH , LOW, STANDARD DEVIATION,ARITHMETIC AVERAGE"
230!"                  AND R FACTOR (MARKET CORRELATION)"
240 !
250!"II.          P/E -- STANDARD DEVIATION"
260 !
270!"III.         VOLUME AVERAGE AND STADARD DEVIATION"
280!
290 !"     IT WILL READ AND WRITE THIS DATA TO AND FROM FILES"
300 !
310 !"***************************************************************"
320 !
330 !TAB (15), "THROUGH-OUT N=NO AND Y=YES"
340 !
350 !TAB(25),"ALL STOCKS HAVE FIVE (5) LETTER NAMES"
360!
370 INPUT "IF YOU ARE READY TO BEGIN , HIT RETURN",Q$
380 GOSUB 1660
390 INPUT "WHAT AVERAGE, DOW OR AMEX? ",E$
400 M=0
410 !
420 IF E$="DOW" THEN 470
430 IF E$="AMEX" THEN 470
440 INPUT "YOU DID NOT ENTER CORRECTLY,HIT RETURN",Q4$
450 GOSUB 1660
460 GOTO 390
470 INPUT "WHAT IS THE NAME OF THE ISSUE? ",S$
480 !
490 REM... THIS ROUTINE GIVES THE AVERAGES
500 GOSUB 1660
510 !
520 !TAB(20),"ANALYSIS OF ",S$," AGAINST ",E$
530 !
540  !
550 REM T=EXC  T1=ISSUE ,   T2= VOLUME,  T3= P/E............................
560 !TAB(11),E$,TAB(24),S$,TAB(38),"VOLUME",TAB(52),"P/E",TAB(64),"% YEILD"
570!
580 IF M=1 THEN 680
590 OPEN #1,E$    \ OPEN #2,S$
600 H5=1
610 READ #1 %5*H5,D
620 IF D=-1 THEN 680
630 READ #2 %20*H5,I,U,P,Y5
640 H5=H5 +1
650 N=N+1 \ T=D+T \ T1=I+T1
660 T2= T2+U \ T3=T3+P \T4=T4+Y5
670 GOTO 610
680 !%#
690 !"MEASRNTS:"     ,TAB(11),N,TAB(25),N,TAB(37),N,TAB (54),N,TAB(68),N
700 CLOSE #1\CLOSE #2
710 !
720 ! %# 8F2
730 !"AVERAGE:",TAB(10),T/N,TAB(23),T1/N,TAB(37),T2/N,TAB(51),T3/N,TAB(64),T4/
740 !
750 A=T/N\ A1=T1/N\ A2=T2/N\ A3=T3/N
760 A4=T4/N
770 IF M=1 THEN 790
780 GOSUB 1690
790 !"  HIGH  ",TAB(10),H,TAB(23),H1,TAB(37),H2,TAB(51),H3,TAB(64),H4
800 !"  LOW",TAB(10),L,TAB(23),L1,TAB(37),L2,TAB(51),L3,TAB(64), L4
810 REM.... THIS IS WHERE WE FIND THE VARIANCES
820 !
830 RESTORE
840 IF M= 1 THEN 1000
850 H5 =1
860 OPEN #1,E$   \OPEN #2,S$
870 READ #1 %5*H5,D
880 IF D=-1 THEN 950
890 READ #2 %20*H5,I,U,P,Y5
900 H5=H5+1
910 W=(D-A)^2+W \ W1=(I-A1)^2+W1
920 W2=(U-A2)^2+W2 \ W3=(P-A3)^2+W3
930 W4=(Y5-A4)^2+W4
940 GOTO 870
950 V=W/N \V1=W1/N \V2=W2/N\ V3=W3/N
960 V4=W4/N
970 CLOSE #1\CLOSE #2
980 S=SQRT(V)\S1=SQRT(V1)\ S2=SQRT(V2)\ S3=SQRT(V3)
990 S4=SQRT(V4)
1000 !        "ST. DV.:",    TAB(10),S,TAB(23),S1,TAB(37),S2,TAB(51),S3,TAB(64)
1010 REM... THIS IS THE CORRELATIONSECTION
1020 !
1030 RESTORE
1040 IF M=1 THEN 1170
1050 OPEN #1,E$\ OPEN #2,S$
1060 H5=1
1070 READ #1 %5*H5,D
1080 IF D =-1 THEN 1150
1090 READ #2 %20*H5,I,U,P,Y5
1100 H5=H5+1
1110 Y=((D-A)*(I-A1))+Y
```

```
1120 X=Y/N
1130 REM
1140 GOTO 1070
1150 R=X/(S*S1)
1160 CLOSE #1\CLOSE #2
1170 !"R FACTOR:",TAB(25),R,",        % UNEXPLAINED: ",100-(R^2*100)
1180 REM... THIS COMPUTES THE REGRESSION LINE
1190 B=X/V
1200 A5=A1-(B*A)
1210 !
1220 !TAB(7),"REGRESION EQUATION: ", 'D = ",A5," + ",B," I"
1230 !
1250 INPUT "DO YOU WANT TO SEE A SPECIFIC VALUE FROM REGRES-EQUATION? ",Q$
1260 !
1270   IF Q$="N" THEN 1320
1280       IF Q$="Y" THEN 1340
1290 INPUT "INCORRECT ENTRY , TRY AGAIN, HIT RETURN.",Q3$
1300 !
1310 GOTO 1250
1320 GOSUB 2760
1330 GOTO 1400
1340 INPUT"WHAT 'I' VALUE? ";I1
1350 !
1360 D1=(I1-A5)/B
1370 !TAB(5),E$,"   ",D1
1380 INPUT "MORE ? ",Q$
1390 IF Q$="Y" THEN 1340
1400 GOTO 2760
1410 M=1
1420 IF Q$="N" THEN 1440
1430      GOTO 490
1440 REM ........BEGIN FILES...."
1450 OPEN #3,"STOCK"
1460 H5=1
1470 READ #3 %55*H5,S1$,X3,X4,X5,X6,X7,X8,X9,W7,W9
1480   IF S1$=S$ THEN 1510
1490 H5=H5+1
1500 GOTO 1470
1510 WRITE #3 %55*H5,S$,A,S,H1,L1,A1,S1,R,    A2,A4
1520 GOSUB 1660
1530 !\!"HERE ARE THE FILED PARAMETERS FOR ",S$," :"
1540 !\!"      DOW AVERAGE:              ",A
1550 !\!"      STANDARD DEVIATION-DOW: ",S
1560 !\!"        HIGH:                   ",H1
1570 !\!"        LOW:                    ",L1
1580 !\!"      STOCK AVERAGE:            ",A1
1590 !\!"      STANDARD DEVIATION:       ",S1
1600 !\!"      CORRELATION:              ",R
1610 !\!"      VOLUME AVERAGE:           ",A2
1620 !\!"      % YIELD AVERAGE:          ",A4
1630 !
1640 CLOSE #3
1650 GOTO 1400
1660 !CHR$(27),"*"
1670 FOR Q=1 TO 4\!CHR$(30)\NEXT
1680RETURN
1690 REM... THIS IS THE SORT ROUTINE
1700 IF M1=1 THEN 1720
1710 DIM D(100 )\ DIMI(100 )\DIM U(100 ) \DIM P(100 )\DIM Y5(100)
1720 RESTORE
1730 OPEN #1,E$\OPEN #2,S$
1740 H5=1\J=1
1750 READ #1 %5*H5,D(J)
1760 IF D(J)=-1 THEN 1820
1770 READ #2 %20*H5,I(J),U(J),P(J),Y5(J)
1780 H5=H5+1
1790 J=J+1
1800 J1=J
1810 GOTO 1750
1820 FOR J=2 TO J1-1
1830 IF D(Q1) <=D(J) THEN 1850
1840 G = Q1   \ Q1  =J       \ J=G
1850 NEXT
1860 FOR J=2 TO J1
1870  IF D(Q2)>=D(J) THEN 1890
1880    G=Q2 \ Q2=J \ J=G
1890  NEXT
1900 H=D(Q2  ) \ L=D(Q1)
1910 CLOSE #1\CLOSE #2
1920 FOR J=2 TO J1-1
1930   IF I(Q3) <=I(J) THEN 1950
1940       G1=Q3   \ Q3  =J       \ J     =G1
1950 NEXT
1960 FOR J=2 TO J1-1
1970   IF I(Q4)>=I(J) THEN 1990
1980   G1=Q4 \ Q4=J \ J=G1
1990 NEXT
2000 H1=I(Q4) \ L1=I(Q3)
2010 REM THIRD VARIABLE
2020    FOR J=2 TO J1-1
2030       IF U(Q5) <=U(J) THEN 2050
2040   G2=Q5 \Q5=J\ J=G2
2050 NEXT
2060 FOR J=2 TO J1-1
2070   IF U(Q6)>= U(J) THEN 2090
2080    G2=Q6 \Q6=J \ J=G2
2090 NEXT
2100 H2=U(Q6) \ L2=U(Q5)
2110 REM PART FOUR-----------
2120   FOR J=2  TO J1-1
2130      IF  P(Q7) <=P(J) THEN 2150
2140        G3=Q7    \ Q7  =J       \ J     =G3
2150 NEXT
2160 FOR J=2 TO J1-1
2170   IF P(Q8)>=P(J) THEN 2190
2180      G3=Q8\Q8=J\J=G3
2190 NEXT
2200 H3=P(Q8) \ L3=P(Q7)
2210 REM PART FIVE................,<<<<<<<<<,,,
2220   FOR J=2 TO J1-1
2230      IF Y5(Q9) <=Y5(J) THEN 2250
2240 G4=Q9    \ Q9   =J         \        J=G4
2250 NEXT
2260 FOR J=2 TO J1-1
```

tween two variables. To illustrate: if we have two separate sets of data and if x increases by five, what does y do? Do the two sets of data move independently of each other (randomly)

Figure 1



Normal Density function:

$$\frac{1}{S\sqrt{2\pi}}\, e^{-(1/2)\,[(X-A)/S]^{2}}$$

A = mean (average), S = standard deviation

The rule of thumb we have been using is seen to be based in the normal distribution function.

or is there a correlation? If there is a correlation is it positive (x varies with y) or negative (x varies inversely with y)?

MARKA Selected Variables

A = index average, A1 = price average, A2 = volume average, A3 = P/E average, A4 = % yield average

D = data values for index

E$= name of index file accessed

H = high value index, H1 = high value price, H2 = high value volume. H3 = high value P/E, H4 = high value % yield, H5 = file counter

I = price of stock from file

L = low value index, L1 = low value price, L2 = low value volume, L3 = low value P/E, L4 = low value % yield

M = switch, M1 = switch

N = number of measurments

P = P/E ratio data from file

R = correlation coefficient

S = standard deviation index, S1 = st. dv. price, S3 = st. dv. P/E, S4 = st. dv. % yield, S$ = name of corporation file accessed

T (1-4) = sums of the parameters

U = volume

V (1-4) = variances of parameters

Y5 = % yield

To find the coefficient of correlation, divide the covariance of the two sets of data by the product of their standard deviations. The formula is given below.

$$R = \frac{\frac{1}{N-1}\sum (x-A)(y-A1)}{\sqrt{\frac{1}{N-1}\sum (x-A)^2}\sqrt{\frac{1}{N-1}\sum (y-A1)^2}}$$

R is the coefficient of correlation, x and y are individual data, A and A1 are the averages of x and y, N is the number of measurements. Note that R is always between -1 and +1. Also, as a rule of thumb: if |R| is greater than 0.81, the odds are good that a correlation does indeed exist. Remember: if R is positive, then as x moves so does y; if R is negative, x moves opposite to y. It's important to understand that the value of the correlation coefficient does not depend on which variable we call x or y. (It is important later in regression analysis, however.)

A note of caution must be added for anyone attempting to infer a cause and effect relationship between two sets of data. A high correlation does not imply that one "controls" or influences the other; they may both be affected by a third or fourth controlling factor. Also, there's no assurance that the relationship of the past will continue into the future.

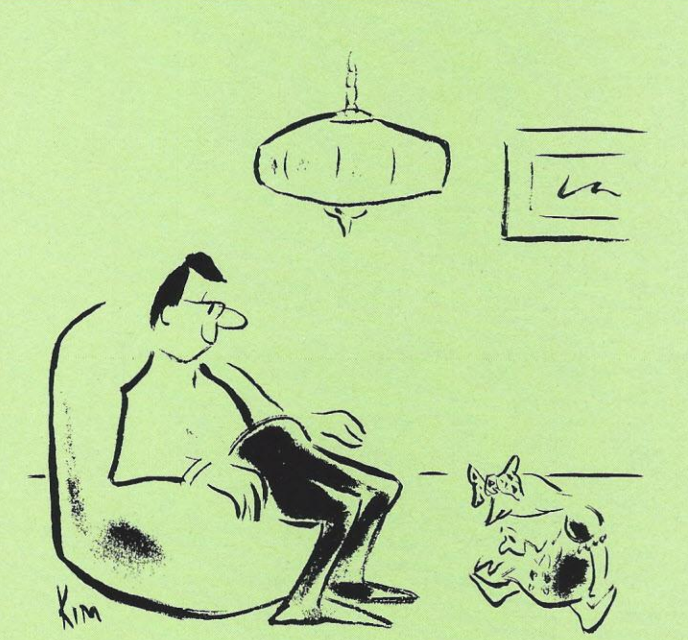
A second quantity directly relates to the coefficient of correlation: the coefficient of deter-

Selected Routines

Initialize variables:	10- 120
Input file names:	370- 480
Compute averages:	580- 760
Compute standard deviation:	830-1000
Compute correlation:	1010-1170
Compute regression line:	1180-1220
File processed parameters:	1440-1680
Sort routine for raw data:	1690-2310
Graph:	2330-2745
Jump table:	2760-2850

Marka Program Listing *– continued*

```
2270   IF Y5(E9)>=Y5(J) THEN 2290
2280      G4=E9 \E9=J\J=G4
2290 NEXT
2300 H4=Y5(E9) \ L4=Y5(Q9)
2310 RETURN
2320 REM
2330 OPEN #2,S$
2340 REM THIS BEGINS THE GRAPH.................................
2350 !%#
2360 GOSUB 1660
2370 H5=1
2380 GOTO 2500
2390 V7=H1-L1
2400 V9=V7/10
2410 V7=     L1 +V7
2420 FOR W9 = 11 TO 1 STEP -1
2430 !"    :"
2440 !INT(V7)," -"
2450 V7=V7-V9
2460 NEXT
2470 !"       ---------|---------|---------|---------|---------|---------|----
2480 !"            10        20        30        40        50        60  WEEK:
2490 GOTO 2620
2500 H9=H1  \L9=L1
2510 V8=    (H1)-    (L1)
2520 V8=INT(V8)
2530 K9 = 1
2540 IF V8> 88 THEN K9 =2
2550 IF V8 > 88 THEN V8 =V8/2
2560 IF V8 > 44 THEN K9 =K9*2
2570 IF V8 > 44 THEN V8 = V8/2
2580 IF V8 > 22 THEN K9= K9*2
2590 IF V8 > 22 THEN V8 =V8/2
2600 I9=    (V8/21)
2610 GOTO 2390
2620 READ #2%20*H5,I,U,P,Y5
2630 IF I=-1 THEN 2730
2640 I =INT (1/K9)-INT(L1/K9)
2650 IF I=0 THEN I=1
2660 T8=INT( I/I9)
2670 T8 = 22 -T8
2680 IF T8<1 THEN T8=1
2690 !CHR$(27),"=",CHR$(31+T8   ),CHR$(31+H5+6) ,"X",
2700 !" "
2710 H5=H5+1
2720 GOTO 2620
2730 !CHR$(27),"=",CHR$(31+24),CHR$(31+1),
2740 !%# 8F2
2745 CLOSE #2
2760 INPUT "WHAT DO YOU WISH TO DO NEXT? ",Q$
2770 IF Q$(1,1)="G" THEN 2330
2780 IF Q$(1,1)="C" THEN M=1
2790 IF Q$(1,1)="C" THEN 490
2800 IF Q$(1,1)="F" THEN 1450
2810 IF Q$(1,1)="R" THEN 2860
2820 IF Q$(1,1)="R" THEN 20
2830 IF Q$(1,1)="E" THEN END
2840 !\!"INCORRECT ENTRY----TRY AGAIN-HIT RETURN. ",Q$
2850 GOTO 2760
2860 CLOSE #1 \CLOSE #2
2870 M1=1 \ GOTO 20
READY
```



"An this little piggy made a killing on the stock market by investing in Computeronics, Inc."

Marka Program Run

```
                         MARKET ANALYSIS PROGRAM

        THIS PROGRAM WILL ANALIZE DATA AND YEILD THE FOLLOWING:

I.          HIGH , LOW, STANDARD DEVIATION,ARITHMETIC AVERAGE
                AND R FACTOR (MARKET CORRELATION)

II.         P/E -- STANDARD DEVIATION

III.        VOLUME AVERAGE AND STADARD DEVIATION

    IT WILL READ AND WRITE THIS DATA TO AND FROM FILES

*****************************************************************

                THROUGH-OUT N=NO AND Y=YES

                         ALL STOCKS HAVE FIVE (5) LETTER NAMES

IF YOU ARE READY TO BEGIN , HIT RETURN
*


WHAT AVERAGE, DOW OR AMEX? DOW

WHAT IS THE NAME OF THE ISSUE? USSTL
*


WHAT AVERAGE, DOW OR AMEX? DOW
*
WHAT IS THE NAME OF THE ISSUE? DUPNT


                  ANALYSIS OF DUPNT AGAINST DOW

           DOW          DUPNT         VOLUME        P/E       % YEILD

MEASRNTS:  52            52           52            52           52

AVERAGE:   893.38       119.97        1719.08       12.48        4.21

  HIGH     980.00       133.75        3371.00       14.00        4.70
  LOW      808.30       107.00         914.00       10.00        3.70

ST. DV.:    49.79         7.23         617.88        1.22         .24

R FACTOR:                     .73,      % UNEXPLAINED:   46.36

     REGRESION EQUATION: D =    25.00 +      .11 I

DO YOU WANT TO SEE A SPECIFIC VALUE FROM REGRES-EQUATION? YE

INCORRECT ENTRY , TRY AGAIN, HIT RETURN.Y

DO YOU WANT TO SEE A SPECIFIC VALUE FROM REGRES-EQUATION? Y

WHAT 'I' VALUE? 100

    DOW     705.49
MORE ? Y
WHAT 'I' VALUE? 130

    DOW     987.69
MORE ? N
WHAT DO YOU WISH TO DO NEXT? END
READY


                  ANALYSIS OF USSTL AGAINST DOW

           DOW          USSTL         VOLUME        P/E       % YEILD

MEASRNTS:  52            52           52            52           52

AVERAGE:   893.38        38.47        4778.50        9.08        5.92

  HIGH     980.00        48.50       14865.00       11.00        7.80
  LOW      808.30        28.48         668.00        8.00        4.50

ST. DV.:    49.79         7.08        2832.57         .78        1.12

R FACTOR:                     .94,      % UNEXPLAINED:   10.81

     REGRESION EQUATION: D =   -81.53 +      .13 I

DO YOU WANT TO SEE A SPECIFIC VALUE FROM REGRES-EQUATION? Y

WHAT 'I' VALUE? 40

    DOW     904.80
MORE ? NO
WHAT DO YOU WISH TO DO NEXT? FILE
*
```

Marka Program Run continued on next page

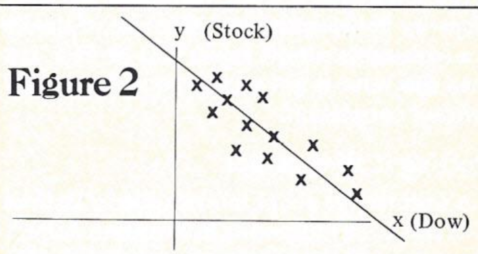
mination. This value is found by squaring the coefficient of correlation: $r=R^2$.

In most cases, R squared is easier to interpret than R because R squared is the percentage of correlation. For example, if R equals 0.5, then R squared is 25% (.5 squared is .25). Thus, we see why the rule of thumb suggested above makes sense; if R equals 0.81, then R squared equals about 66%. Also note, if you subtract R squared from 100 you get the percentage unexplained by the correlation analysis – the randomness between two sets of data. Understand that the degree of correlation is a probability, not a certainty! And even if the correlation is great, again there is no guarantee that this relationship will continue in the future.

The major operation we'll be using is regression line analysis. The regression equation is the line of "best fit" for a given set of data. The regression analysis we will deal with always finds the linear equation.

To find the regression equation employ the "Method of Least Squares Estimation" (See Figure 2). Here is a co-ordinate axis, with X and Y representing two separate parameters, in this case the Dow Jones index and the price of U. S.

Figure 2



Scatter-gram with regression line

Steel. The point of intersection of each individual measurement forms a "scatter-gram" of the two parameters. Your job is to find a line which best fits all the data present, slicing evenly through the points so as to cancel the positive residuals with the negative residuals. To find this equation, first find the slope of the line by dividing the covariance of X and Y by the variance of the independent variable. The formula looks like this:

$$SLOPE = M = \frac{\frac{1}{N-1}\sum (X-A)(Y-A1)}{\frac{1}{N-1}\sum (X-A)^2}$$

Once you have found the slope, you need

only find one point which is on the line to find the equation. Without explaining why here, it can be shown that the regression line must pass through the point (A, A1), the averages of the two parameters. Given this point and the slope, we use the slope-point method of finding the equation.

Perhaps the single most important use of the regression line is to predict future events. Consider the following example. The Dow index is currently at 790, the price per share of U.S. Steel is now at $30, the highest the Dow is expected to go is 890 – can you expect to make money? Second, how sure are you the prediction is correct? To answer these questions, turn to the correlation coefficient and the regression line. With the regression line, you substitute the expected value of the Dow index (in the example) and solve for the expected price of U.S. Steel. We find we can expect to make a profit... if the Dow index actually goes that high and if our regression analysis accurately reflects the real situation.

The correlation coefficient will tell us just how well the data correlates, and in what direction. If the correlation is large, then we know the data and the regression line nearly coincide; if the correlation is small, then there is a wide disparity between the two. Our rule of thumb is a reasonable cut-off value, 0.81 or 66%; less than this value is not usually considered a significant correlation.

Putting it all together

Let's utilize the above information to analyze U.S. Steel. The following information comes from a run of the program included with this article. The entire year 1977 was used for the raw data.

Parameters for U. S. Steel

parameter	average	St. Dv.	High	Low
Price/share	38.47	7.08	48.5	28.48
Volume	4778.5	2823.57	14865	668
P/E	9.08	0.78	11	8
% Yield	5.92	1.12	7.8	4.5
INDEX	893.38	49.79	980	808.3

Correlation: 0.94 Percentage correlation: 89%

There are two things we immediately see – first an excellent correlation with the Dow average, and second, a substantial spread between the

Marka Program Run *– continued*

```
HERE ARE THE FILED PARAMETERS FOR USSTL :

     DOW AVERAGE:                    893.38

     STANDARD DEVIATION-DOW:          49.79

       HIGH:                          48.50

       LOW:                           28.48

     STOCK AVERAGE:                   38.47

     STANDARD DEVIATION:               7.08

     CORRELATION:                       .94

     VOLUME AVERAGE:                4778.50

     % YIELD AVERAGE:                  5.92

WHAT DO YOU WISH TO DO NEXT? REPEAT
```

Marka Subroutines

```
10 REM **** THIS ROUTINE WILL ENTER D TA INTO A CORPORATE DATA FILE
20 REM **** FOR USE WITH THE ANALYSIS PROGRAM
30 REM **** THE DATA GIVEN BELOW IS O LY FOR MMM CORP., AND IS INCLUDED ONLY
40 REM **** AS AN EXAMPLE
50 OPEN #2, "MMM  "
60 H=1
70 FOR Y=1 TO 53
80 READ I,U,P,Y1
90 WRITE #2 %20*H,I,U,P,Y1
100 !I,TAB(10),U,TAB(20),P,TAB(30),Y1,TAB(40),"WEEK # ",H
110 H=H+1
120 NEXT
130 END
140 DATA 52.5,    , 6782, 18, 2.8, 52.75,  6222, 18, 2.8, 52.5,   6782, 18, 2.
150 DATA 51,        4456, 17, 2.8, 50.75,  4387, 17, 2.9, 49.25,  5277, 17, 2.
160 DATA 50.75,     6437, 17, 3.3, 50.75,  6437, 17, 3.3, 52.25,  2741, 18, 3.
170 DATA 52.125,    2428, 18, 3.3, 53.75,  2693, 18, 3.2, 50.875, 2273, 17, 3.
180 DATA 50.625,    2623, 17, 3.4, 50,     2160, 17, 3.4, 50,     2160, 17, 3.
190 DATA 49.125,    3190, 17, 3.5, 48.5,   3119, 17, 3.5, 50.5,   2326, 16, 3.
200 DATA 49.875,    1830, 16, 3.4, 51,     3627, 17, 3.3, 47.5,   4299, 15, 3.
210 DATA 48.75,     2326, 16, 3.5, 47.875, 3992, 15, 3.6, 48.5,   2541, 16, 3.
220 DATA 50,        3143, 16, 3.4, 48.75,  1975, 16, 3.5, 48.875, 2779, 16, 3.
230 DATA 48.625,    2152, 16, 3.5, 50.75,  2635, 16, 3.3, 50.125, 2876, 16, 3.
240 DATA 50,        2202, 16, 3.4, 50.625, 3308, 16, 3.4, 52.25,  4680, 16, 3.
250 DATA 51.625,    2517, 16, 3.3, 52.5,   1954, 16, 3.3, 51.75,  1663, 16, 3.
260 DATA 51,        2248, 16, 3.3, 50,     3049, 16, 3.4, 50.625, 2508, 16, 3.
270 DATA 50.625,    2342, 16, 3.4, 49.25,  3263, 15, 3.5, 46.5,   3884, 15, 3.
280 DATA 48.625,    3520, 14, 3.5, 47.125, 2183, 14, 3.6, 50,     1871, 15, 3.
290 DATA 49.25,     2765, 14, 3.5, 49.25,  2108, 14, 3.5, 47,     3515, 14, 3.
300 DATA 46.625,    3058, 14, 3.6, 46,     3106, 14, 3.7, 47.625, 5177, 14, 3.
310 DATA 48.5,      3798, 14, 3.5
320 DATA -1,-1,-1,-1
READY


LIST

10 REM **** THIS ROUTINE WILL INTITIALIZE THE "STOCK" DATA FILE
20 REM **** THE COMPANIES LIST BELOW ARE ONLY EXAMPLES, OTHERS MAY BE USED
30 OPEN#3,"STOCK"
40 FOR X=1 TO 51
50 READ A$
60 WRITE #3 %60*X,A$,A,S,D,F,G,H,J,K,K1,J5
70 NEXT
80 CLOSE #3\END
90 DATA"EXXON","GNMOT","MOBIL","TEXAC","IBM  ","FORDM","STOIL","GULFO"
100 DATA"SEARS","GENEL","STOND","ITT  ","USSTL","ATRCH","SHELL","TENNE","CHRY
110 DATA"DUPNT","DOWCH","UNCRB","STOOH","CNTOL","ESKOD","WESTH","WESTE","PHPE
120 DATA"BTHST","SUNCO","XEROX","GTE  "
130 DATA "UNNCL"
140 DATA "CAPTR", "MONSA" ,"REYND", "MMM  ", "BANKA", "CONED","INTPP"
150 DATA "BURRO", "AMEXP", "CBS  ","RCA  ", "UNTCH","PPGIN", "CGPAL"
160 DATA "TLDYN", "AMCYD","TRW  ", "ALDCH", "RCKWL","GYEAR","TXTRN"
READY


LIST

10  REM **** THIS IS A EXAMPLE OF HOW TO LOAD DATA INTO AN 'INDEX' FILE
20 OPEN #2,"DOW"
30   FOR X=1 TO 39
40      READ D
50 WRITE #2 %5*X,D
60 NEXT
70 CLOSE #2
80 DATA 980, 967,962.43, 957.53, 947.89,931.52,940.24,933.5, 953.46
90 DATA 947.72,961.02, 928.86
100 DATA 927.36, 918.88,944, 927.07, 926.9, 936.74, 928.34, 930.46
110 DATA898.83, 912.23, 910.79, 920.45, 929.7, 912.65,907.99, 905.95,921
130 DATA 890, 888.69, 871.1, 863.48,855.42, 872.31, 857.07, 856.81, 839.14
140 DATA -1
READY
```

With the three routines listed above, it should be easy to initialize, enter and load data into any of the three data files outlined in the text. One word of caution, make sure that you do not accidentally enter data out of order, after all it may affect your finances!!

Stock Access Program Listing

```
10 REM **** THIS IS THE SCHILDT STOCK RANDOM ACCESS PROGRAM FOR
20 REM **** PROCCESSED DATA.
 0 REM **** IT IS MEANT FOR USE WITH THIS SCHILDT STOCK MARKET ANALYSIS
40 REM **** PACKAGE. NO RIGHTS ARE GRANTED FOR DUPLICATION.
50 REM
80 REM
85 GOSUB 1500
90 OPEN #1,"STOCK"
100 M=0 \M1=0 \M2=0 \M3=0\ M4=0\ M5=0 \REM SWITCH VARIABLES FOR PARAMETERS
110 N=0 \N1=0 \ REM SWITCH FOR GREATER \LESS THAN FOR ST. DV.
120 W=0 \ W1=0 \REM "                                  "  PRICE/SHARE
130 B=0 \B1=0 \ REM "                                  "  CORRELATION
140 C=0 \ C1=0\ REM "                                  "  VOLUME
150 Z=0 \ Z1=0\ REM "                                  "  P/E
160 L=0 \L1=0 \ REM "                                  "  % YIELD
170 K=0 \ REM      "                                  " NAME CORP ACCESS
180 GOSUB 1400
190 ! \ !"               D O   Y O U   W I S H   T O   A C C E S S   B Y "
200 ! \ INPUT "                C O R P O R A T E   N A M E ? :",Q$
210 GOSUB 1410
220 IF Q$="Y" THEN 250
230 GOTO 280
240 GOSUB 1400
250 !\INPUT "              E N T E R   N A M E   H E R E  : ",S1$
260 K=1
270 GOTO 740
280 GOSUB 1400
 90 !\       !"              D O   Y O U   W I S H   T O   A C C E S S   B Y "
 00 !\ INPUT "               A V E R A G E   P R I C E  /  S H A R E ? ",Q$
310 GOSUB 1410
320 IF Q$="Y" THEN M1=1 ELSE 390
330 INPUT "                        H O W ? ", Q$
340 GOSUB 1410
350 GOSUB 1430
360  INPUT "",K1
370 IF Q$="G" THEN W=1
380 IF Q$="L" THEN W1=1
390 !\INPUT"               C O R R E L A T I O N ? ",Q$
400 IF Q$="Y" THEN M2=1 ELSE 470
410 INPUT"                        H O W ? ",Q$
420 GOSUB 1410
430 IF Q$="G" THEN B=1
440 IF Q$="L" THEN B1=1
450 GOSUB 1430
460 INPUT "",J
470 !\INPUT"               S T A N D A R D   D E V I A T I O N ? ",Q$
480 GOSUB 1410
490 IF Q$="Y" THEN M =1 ELSE 560
500 INPUT"                        H O W ? ",Q$
510 GOSUB 1410
520 IF Q$="G" THEN N=1
530 IF Q$="L" THEN N1=1
540 GOSUB 1430
 50 INPUT "",P
560 !\INPUT "              V O L U M   ? ",Q$
570 GOSUB 1410
580 IF Q$="Y" THEN M3=1 ELSE 650
590 INPUT "                        H O W ? ",Q$
600 GOSUB 1410
610 IF Q$="G" THEN C=1
620 IF Q$="L" THEN C1 =1
630 GOSUB 1430
640 INPUT "",G
650 !\INPUT "              %   Y I E L D  ? ",Q$
660 GOSUB 1410
670 IF Q$="Y" THEN M5 =1 ELSE 740
680 INPUT "                        H O W ?  ",Q$
690 GOSUB 1410
700 IF Q$="G" THEN L=1
710 IF Q$="L" THEN L1=1
720 GOSUB 1430
730 INPUT "",F
740 REM THIS IS THE PRINT-OUT \ SIEVE ROUTINE
750 H=1
760 READ #1 %55*H,S$,A,S,H1,L1,A1,S1,R,A2,A4
770 IF S$="GYEAR" THEN 1450
780 H=H+1
790 IF K=1 THEN 1100
800 IF M1 = 1 THEN 810\ IF M1 = 0 THEN 860 \ REM **** PRICE /SHARE
810 IF W=1 THEN 840
820 IF W1=1 THEN 850
830 IF A1=K1THEN 860 ELSE 760
840 IF A1>K1THEN 860 ELSE 760
850 IF A1<K1 THEN 860 ELSE 760
860 IF M2=1 THEN 870 ELSE 920 \ REM **** R CORRELATION
870 IF B=1 THEN 900
880 IF B1=1 THEN 910
890 IF R=J  THEN 920 ELSE 760
900 IF R> J  THEN 920 ELSE 760
910 IF R< J THEN 920 ELSE 760
920 IF M=1 THEN 930 ELSE 980 \REM **** STANDARD DEVIATION
930 IF N=1 THEN 960
940 IF N1=1 THEN 970
950 IF S1=P THEN 980ELSE 760
960 IF S1>P THEN 980 ELSE 760
970 IF S1<P THEN 980 ELSE 760
980 IF M3=1 THEN 990 ELSE 1040 \REM **** VOLUME
990 IF C=1 THEN 1020
1000 IF C1=1 THEN 1030
1010 IF A2=G THEN 1040 ELSE 760
1020 IF A2>G THEN 1040 ELSE 760
1030 IF A2<G THEN 1040 ELSE 760
1040 IF M5=1 THEN 1050 ELSE 1120 \REM **** YIELD
1050 IF L=1 THEN 1080
1060 IF L1=1 THEN 1090
1070 IF A4=F THEN 1120 ELSE 760
1080 IF A4>F THEN 1120 ELSE 760
1090 IF A4<F THEN 1120 ELSE 760
1100 IF S1$=S$ THEN 1120
1110 GOTO 760
```

Stock Access Program Listing continued on next page

high and low price per share. Those already familiar with common brokerage commissions know it takes a hefty increase to make any great profit. Therefore, the spread between the high and the low are of special importance. Likewise, the standard deviation.

Standard deviation gives us a probability of the value of the stock at any one point in time

Stock Access Program Notes

The random access program enables the user to retrieve statistical information on any corporation or corporations quickly and easily. It is entirely user-prompted. The program requires use of the "stock" data file.

As with all other programs in the market analysis package, this program is written in North Star BASIC. Conversion to other disk-based BASICs should be no problem. It would seem possible to convert to a cassette operating system; however the access time could easily be on the order of minutes instead of seconds. The program is really just a group of software switches with input instructions. If you type in the program, or modify it, I would recommend a thorough test. It's easy to forget to turn a switch on (or off).

This program requires around 20K RAM to run, but much could be saved at the sacrifice of the liberal PRINT statements used in the samples.

To use the program, simply answer the computer prompts. For instance if you wish to access all corporations whose P/E is less than 10, and whose price per share is greater than $30, one would type 'G' '30' when asked about price per share, and 'L' '10' when the system asks about the P/E; the program will do the rest.

If your terminal does not have a "clear screen" command, delete this command in line 1440, substituting a REM or other statement of your choice.

The selected list of variables used in the access program include:

M - M5 =	the variables used as switches for the parameters, i.e., price/share, etc.
N, N1 =	switches for standard deviation
B, B1 =	switches for correlation
C, C1 =	switches for volume
L, L1 =	switches for yield
K =	switch for access by name only

REM statements should provide enough information to understand and modify the program if desired.

falling within a deviation or a multiple of a deviation. Again, this is the probability of *past* occurrences, not necessarily of future prospects. However, *if* there is no fundamental change in the probability distribution, then the odds can be good that a near future price movement will lie within three deviations of the average we found.

The same relates to other parameters in Table 1. Please note that the correlation refers only to the price per share plotted against the Dow average.

A correlation analysis could easily be constructed to find the correlation between any of the parameters. For my own purposes, I found that correlating the Dow index with the price per share was of the most aid to me. Other ways to analyze the market often involve some or all of the other factors in correlation analysis.

Now, with the personal computer, you can compete with, and even beat, the corporate analysts at their own game!

In fact, it's the computer's power that allows us the freedom of experimentation in market research – we can construct and test a theory without hours of laborious calculation. Feel free to improvise and improve this or any market analysis system.

Back to our example. To decide whether U.S. Steel could be profitable, we turn to the regression line. The chart tells us that U.S. Steel tends to move with the Dow index; that is, when the Dow goes up, U.S. Steel goes up and vice versa. Also the correlation is strong (greater than our 66% rule). Hence, we can expect the regression line and the scatter-gram of U.S. Steel to coincide closely. We can "plug" values into the regression equation and expect our results to reflect reality.

Let's experiment for a while: If, let's say, the price of U.S. Steel is at $30 per share, how much can you hope to make if the correlation between

Stock Access Listing *– continued*

```
1120 REM **** THIS PRINTS THE CHART
1130 GOSUB 1400
1140 !\!
1150 !
1160 !TAB(15),'CORPORATION:                '‚S$
1170 !
1180 !TAB(15),'DOW AVERAGE:                '‚A
1190 !
1200 !TAB(15),'STANDARD DEVIATION-DOW:     '‚S
1210 !
1220 !TAB(15),'AVERGAE PRICE/SHARE:       $'‚A1
1230 !
1240 !TAB(15),'HIGH:                       '‚H1
1250 !
1260 !TAB(15),'LOW:                        '‚L1
```

Stock Access Program Run

```
            THROUGH-OUT USE YES, NO, GREATER THAN, LESS THAN
            ---ALSO, D O  N O T  U S E  D O L L A R  S I G N S --
-
TYPE 'YES' AND RETURN TO BEGINYES
*
            D O  Y O U  W I S H  T O  A C C E S S  B Y
                 C O R P O R A T E  N A M E ? :YES
         E N T E R  N A M E  H E R E : SEARS
*

            CORPORATION:                SEARS
            DOW AVERAGE:                 893.3825
            STANDARD DEVIATION-DOW:      49.789967
            AVERGAE PRICE/SHARE:       $ 46.101442
            HIGH:                        64.625
            LOW:                         27.875
            STANDARD DEVIATION-SEARS      15.309508
            CORRELATION:                 .90781091
            VOLUME-AVERAGE:              5816.1154
            YIELD-AVERAGE:             % 3.0480769
M O R E ? YES
*
            D O  Y O U  W I S H  T O  A C C E S S  B Y
                 C O R P O R A T E  N A M E ? :NO
*
            D O  Y O U  W I S H  T O  A C C E S S  B Y
            A V E R A G E  P R I C E / S H A R E ? YES
              H O W ? GREATER
            E N T E R  P A R A M E T E R : 30
            C O R R E L A T I O N ? NO
            S T A N D A R D  D E V I A T I O N ? NO
            V O L U M E ? NO
            %  Y I E L D  ? YES
              H O W ?  GREATER
            E N T E R  P A R A M E T E R : 6
*

            CORPORATION:                FORDM
            DOW AVERAGE:                 893.3825
            STANDARD DEVIATION-DOW:      49.789967
            AVERGAE PRICE/SHARE:       $ 56.465705
            HIGH:                        60.75
            LOW:                         41.875
            STANDARD DEVIATION-FORDM      8.4281401
            CORRELATION:                 .52954207
            VOLUME-AVERAGE:              5924.8526
            YIELD-AVERAGE:             % 7.5935897
M O R E ? YES
*
```

```
CORPORATION:                 GTE

DOW AVERAGE:                  893.3825

STANDARD DEVIATION-DOW:       49.789967

AVERGAE PRICE/SHARE:        $ 31.125962

HIGH:                         38

LOW:                          24.125

STANDARD DEVIATION-GTE         1.0845723

CORRELATION:                  -.37246511

VOLUME-AVERAGE:               3715.5385

YIELD-AVERAGE:              % 6.7807692

M O R E ? NO
D O N E   W I T H   S E A R C H  :  D O   I T   A G A I N  ?NO
```

Stock Access Subroutines

```
10 REM **** THIS ROUTINE WILL ENTER DATA INTO A CORPORATE DATA FILE
20 REM **** FOR USE WITH THE ANALYSIS PROGRAM
30 REM **** THE DATA GIVEN BELOW IS ONLY FOR MMM CORP., AND IS INCLUDED ONLY
40 REM **** AS AN EXAMPLE
50 OPEN #2, 'MMM  '
60 H=1
70 FOR Y=1 TO 53
80 READ I,U,P,Y1
90 WRITE #2 %20*H,I,U,P,Y1
100 !I,TAB(10),U,TAB(20),P,TAB(30),Y1,TAB(40),'WEEK # ',H
110 H=H+1
120 NEXT
130 END
140 DATA 52.5,    , 6782, 18, 2.8, 52.75,  6222, 18, 2.8, 52.5,   6782, 18, 2.8
150 DATA 51,        4456, 17, 2.8, 50.75,  4387, 17, 2.9, 49.25,  5277, 17, 2.9
160 DATA 50.75,     6437, 17, 3.3, 50.75,  6437, 17, 3.3, 52.25,  2741, 18, 3.3
170 DATA 52.125,    2428, 18, 3.3, 53.75,  2693, 18, 3.2, 50.875, 2273, 17, 3.3
180 DATA 50.625,    2623, 17, 3.4, 50,     2160, 17, 3.4, 50,     2160, 17, 3.4
190 DATA 49.125,    3190, 17, 3.5, 48.5,   3119, 17, 3.5, 50.5,   2326, 16, 3.4
200 DATA 49.875,    1830, 16, 3.4, 51,     3627, 17, 3.3, 47.5,   4299, 15, 3.6
210 DATA 48.75,     2326, 16, 3.5, 47.875, 3992, 15, 3.6, 48.5,   2541, 16, 3.5
220 DATA 50,        3143, 16, 3.4, 48.75,  1975, 16, 3.5, 48.875, 2779, 16, 3.5
230 DATA 48.625,    2152, 16, 3.5, 50.75,  2635, 16, 3.3, 50.125, 2876, 16, 3.4
240 DATA 50,        2202, 16, 3.4, 50.625, 3308, 16, 3.4, 52.25,  4680, 16, 3.3
250 DATA 51.625,    2517, 16, 3.3, 52.5,   1954, 16, 3.3, 51.75,  1663, 16, 3.3
260 DATA 51,        2248, 16, 3.3, 50;     3049, 16, 3.4, 50.625, 2508, 16, 3.4
270 DATA 50.625,    2342, 16, 3.4, 49.25,  3263, 15, 3.5, 46.5,   3884, 15, 3.7
280 DATA 48.625,    3520, 14, 3.5, 47.125, 2183, 14, 3.6, 50,     1871, 15, 3.4
290 DATA 49.25,     2765, 14, 3.5, 49.25,  2108, 14, 3.5, 47,     3515, 14, 3.6
300 DATA 46.625,    3058, 14, 3.6, 46,     3106, 14, 3.7, 47.625, 5177, 14, 3.6
310 DATA 48.5,      3798, 14, 3.5
320 DATA -1,-1,-1,-1
READY


LIST  .

10 REM **** THIS ROUTINE WILL INTITIALIZE THE 'STOCK' DATA FILE
20 REM **** THE COMPANIES LIST BELOW ARE ONLY EXAMPLES, OTHERS MAY BE USED
30 OPEN#3,'STOCK'
40 FOR X=1 TO 51
50 READ A$
60 WRITE #3 %60*X,A$,A,S,D,F,G,H,J,K,K1,J5
70 NEXT
80 CLOSE #3\END
90 DATA'EXXON','GNMOT','MOBIL','TEXAC','IBM  ','FORDM','STOIL','GULFO'
100 DATA'SEARS','GENEL','STOND','ITT  ','USSTL','ATRCH','SHELL','TENNE','CHRYS'
110 DATA'DUPNT','DOWCH','UNCRB','STOOH','CNTOL','ESKOD','WESTH','WESTE','PHPET'
120 DATA'BTHST','SUNCO','XEROX','GTE  '
130 DATA 'UNNCL'
140 DATA 'CAPTR', 'MONSA' ,'REYND', 'MMM  ', 'BANKA', 'CONED','INTPP'
150 DATA 'BURRO', 'AMEXP', 'CBS  ','RCA  ', 'UNTCH','PPGIN', 'CGPAL'
160 DATA 'TLDYN', 'AMCYD','TRW  ', 'ALDCH', 'RCKWL','GYEAR','TXTRN'
READY


LIST

10  REM **** THIS IS A EXAMPLE OF HOW TO LOAD DATA INTO AN 'INDEX' FILE
20 OPEN #2,'DOW'
30   FOR X=1 TO 39
40     READ D
50 WRITE #2 %5*X,D
60 NEXT
70 CLOSE #2
80 DATA 980, 967,962.43, 957.53, 947.89,931.52,940.24,933.5, 953.46
90 DATA 947.72,961.02, 928.86
100 DATA 927.36, 918.88,944, 927.07, 926.9, 936.74, 928.34, 930.46
110 DATA898.83, 912.23, 910.79, 920.45, 929.7, 912.65,907.99, 905.95
120 DATA 921
130 DATA 890, 888.69, 871.1, 863.48,855.42, 872.31, 857.07, 856.81, 839.14
140 DATA -1
READY
```

These three routines make initializing entering and loading data into any of the three data files outlined in the text easy. One word of caution. Make sure that you do not (accidentally) enter data out of order – it may affect your finances!!

U.S. Steel and the Dow remains the same and the Dow goes to 900?

By solving the regress equation we find the value of U.S. Steel is likely to be at approximately $40 per share.

Let's view it from another angle. If I'm contemplating the purchase of U.S. Steel (whose current price is $30 per share with the Dow at 830), how far does the Dow have to go up for me to make a profit? Again, solving the regression equation, we find that to experience a $10 pre-tax, pre-commission profit, the Dow will have to go to approximately 900.

Therefore, if you think it's reasonable for the Dow to go that high in a "reasonable" length of time, then you can feel secure in your purchase. However, if this rise doesn't seem reasonable, then don't buy.

Along similar lines, if you feel the maximum downside risk you can accommodate is the loss

The better informed the investor is, the greater the likelihood of success. If the figures don't frighten you, buy; if they do, pass.

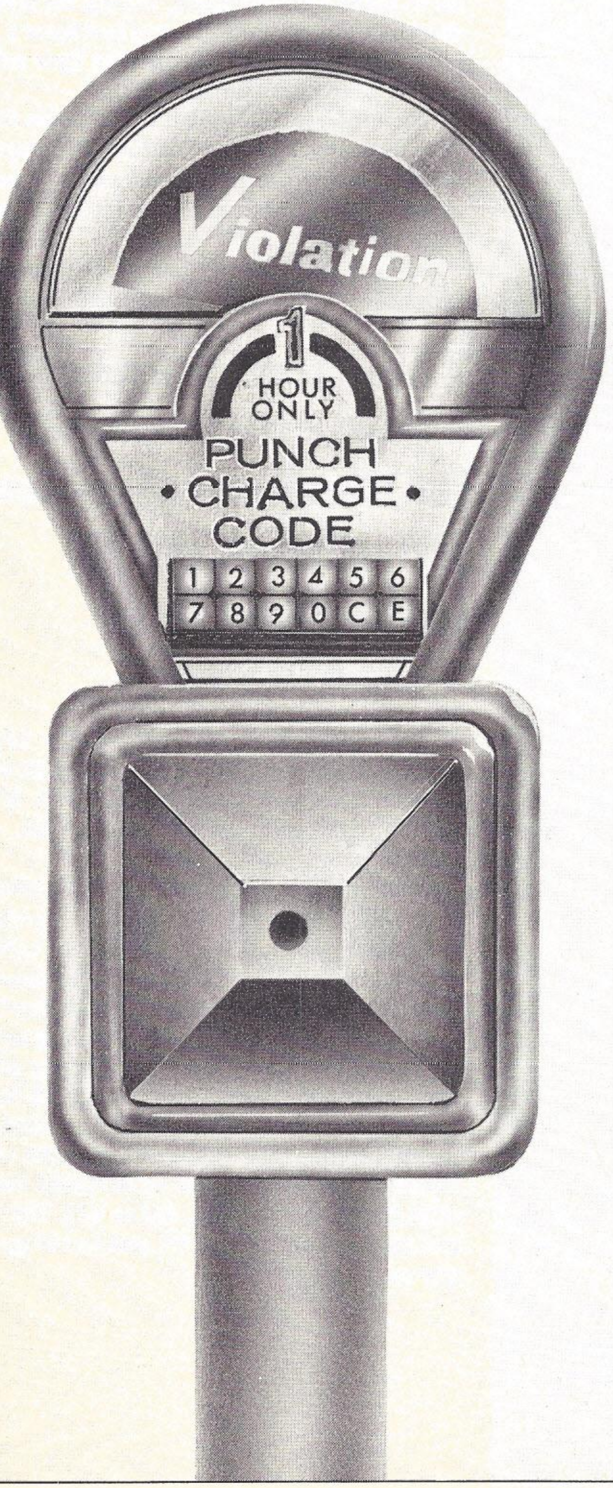
of $5 per share, then "plug" $25 ($30 - $5) into the regress equation, and find that the Dow would have to go to approximately 790. If this doesn't frighten you, buy; if it does, pass. (That's the way I operate.)

Remember, if the correlation is not greater than 66%, you shouldn't rely on the regression analysis. However, if there is sufficient correlation, often (not always) results can be improved. Also, it's good to compare entire industry blocks to individual companies to see which, if any, are the most sensitive to the movements of the Dow, and in what way. Be sure to check for intervening variables, such as an entire industry group suffering. Watch out for industries or companies on a permanent or stretched out decline (i.e., A & P, airlines, and so on). They may not recover soon enough for you to recognize any real profit.

ELECTRONIC FUND TRANSFERS

A Promise or a Threat?

BY H. PARIS BURSTYN



With microcomputer-based remote terminals in stores and on street corners, computerized banking will someday provide more convenient banking services than banks currently offer. They'll be the same size as, and have the same intelligence level as, most home computers. And they'll be tied to large banks' central computers to provide instant loan authorizations, credit and check guarantees and perhaps even instantaneous money transfers from customer to merchant accounts (in order to replace, or at least supplement, cash transactions).

But these computer networks (you might even be able to tie your home computer to the bank's, through a specialized modem) could pose a threat to privacy. Large amounts of personal data, kept in some electronic form, will be subject to unauthorized access. Not that today's manual systems are not subject to similar breaches, but in some aspects, an electronic system would make privacy invasions easier.

Some banking and government sources strongly believe electronic payment systems will threaten personal privacy and bank security, while other sources believe the new technologies will provide greater security, both for banks and individuals. Pointing to the relative security of the few existing systems allows electronic banking proponents to buttress their viewpoints.

Still, other reports on computer crime point out that computer security system breaches are next to impossible to detect (the FBI doesn't even try to detect computer crime; it will only investigate reported crimes to gather evidence for prosecution).

A recent article in *New Yorker* magazine described a number of computer crimes where only a manual mistake or sloppiness betrayed the thief (one bank teller, who "borrowed" close to $30,000 a day to support his gambling, was caught after his bookie was arrested).

Electronic banking has been around for quite some time, in areas not directly visible to the public. But as the technology becomes less expensive and more expansive, it will affect more and more people. During the past 20 years computerized banking operations increased significantly in complexity. For quite some time the entire Federal Reserve System has depended on electronic fund transfers between member banks and the Treasury Department. More recently, banks and other "depository institutions" began offering computerized services to supplement checks and currency. Some social security recipients and employees have their checks deposited electronically to their accounts while some consumers pay bills by telephone. These services and the technology that facilitates them are called electronic funds transfer (EFT).

EFT means something different to everyone. To bankers, EFT represents a way to cut costs and increase efficiency. To the public EFT means more convenient banking hours and last-minute payment so funds can collect maximum interest. To the Government, it's a way to speed grant, social security and other Federal payments. And to many individuals, EFT poses a potential threat to privacy.

In 1976, Americans wrote approximately 28 billion checks against approximately 106 million checking accounts. The volume of checks written grows about 7% each year. These figures represent, in part, the incredible

Illustration by David Gardner

CRT/TERMINAL

Fully Assembled And Tested ! ! !

$399.^{00}

STANDARD FEATURES:

16 x 32 SWITCHABLE TO 64 x 16 DISPLAY

RS232 I/O PORT

GUARANTEED FOR 90 DAYS

OPTIONS:

- **9" OR 15" MONITOR**
- **NUMERIC KEY PAD**
- **SPECIAL FUNCTION KEY PAD**
- **MODEM**
- **S100 BUS BACKPLANE**
- **PLUS MANY MORE - CIRCLE BINGO NUMBER FOR MORE DETAILS**

Caldwell Computer Co.

546 West Olney Avenue
Philadelphia, PA 19120
(215) 224-9133

Sam Caldwell

CIRCLE 17

amount of paper U.S. banks must handle every year from checking accounts alone. Aside from checking account paper, another paper trail follows savings withdrawals, savings deposits, loan slips, loan payments and various other "internal documents".

Banking sources say EFT will reduce the growth of these internal documents as new accounts become computerized.

"What the banking system is trying to do for the consumer, we've been doing for corporations for 30 years – ever since we had a teletype or telephone," said Eugene M. Tangney, a senior vice president of the First National Bank of Boston, who, with Richard Hill, First National's chairman of the board, served on the National Commission on Electronic Funds Transfers (NCEFT). (See Box.)

"Presently, a corporate treasurer can move, deposit, withdraw, make loan payments and make loan drawdowns, all over the United States by wire," Tangney continued. "We're trying to take that corporate system and bring it to the consumer."

In general terms, EFT can provide four consumer services: 1) Automatic deposits to, and payments from, personal accounts 2) Widespread branch banking 3) Check verification and 4) Debit cards.

In many places, banks already offer automatic deposit and payment services. But presently, only a few states have liberal enough banking laws to permit widespread branch banking. In most states, banking laws, designed to protect small institutions, restrict the extent to which banks may set up branches. A controversy currently rages in financial and government circles over the definition of whether "automated teller machines" (ATMs – where customers put in their plastic card and then cash checks, deposit money or transfer money from savings to checking or vice versa) constitute "brick and mortar" branches.

Some banks have sidestepped the issue by building ATMs into the walls of their existing offices to allow customers 24-hour (or at least longer than regular banking hours) access to their accounts. It's all a beginning in the trend toward electronic banking.

In New England and some other parts of the country, many banks now cooperate with supermarkets to provide customers with check cashing facilities. Some facilities are based on "manual systems", ID cards issued by the store, while others employ electronic systems with charge card-type cards that access bank verification information in an online computer environment.

Debit cards – used at stores in place of cash or checks for instantaneous transfers of customer money to store accounts – exist only in experimental systems.

Tangney at Boston's First National Bank said there's "no way" debit cards would replace cash and check transactions. But sometime in the future, he thinks, debit cards will serve as an alternative to cash and checks for payments. "Until the costs come down," he said, "widespread (terminal) installation will be difficult."

My object is to assure there is no problem, but I wouldn't say there is no problem . . ."

But will the public and the banking institutions rush to jump on the EFT bandwagon? EFT will mean adapting to an entirely new way of doing business. And how many consumers are dissatisfied with the way today's system works?

"Although EFT is not difficult to understand, it's going to take a great deal of education before people fully grasp the concept," Tangney said. "The biggest problem is they don't understand the way the present (paper based) system works – so it's difficult to make comparisons."

Stamford, CT-based financial consultant Deborah Rechnitz said, "I think people are willing to make that change (from the paper-based system to EFT) with a little bit of prodding, and maybe a little bit of marketing."

And while she acknowledged that education is part of the bank's marketing plan, she said, "The bank is not going to tell the person what he's giving up – real control and knowledge of his financial position. And that's the biggest danger, not knowing how much money you've got."

Without the paper-based receipts we presently rely on to record our financial position, many of us would lose touch with our financial situation.

But while some people argue that we're comfortable with existing EFT services like electronic pay check and social security deposits, they're ignoring that we still receive pay stubs through the mail a day or two after an electronic deposit.

One EFT system manufacturer will offer consumers POS terminals with three different record keeping methods. One terminal requires customers to ask for their balance after completing a transaction and does not print a receipt. Another terminal produces a receipt of the transaction, but doesn't tell you what your balance was before or after the transaction. And a third would print your balance onto the transaction document after the transaction.

Financial consultant Rechnitz believes people will become more and more comfortable with fewer and fewer receipts as time goes on. "The average consumer doesn't like handling his personal finances. If someone else (a bank, for instance) is willing to take over that responsibility, he'll gladly give it up," she said.

If and when this transfer of responsibility takes place, the consumer will be saying, in effect, "Here are my bills, you pay them when they come due, send me my statement and let me know how I'm doing."

The average individual believes banks rarely make mistakes. It's a matter of trust. How often is your statement wrong? And how often is it actually the bank's fault? Rarely, in most individuals' experience. The majority of people receive their statements and just assume they're right. If their checkbook doesn't balance they simply say, "Well, I made a mistake somewhere," and they "correct" their balance to match the bank's and forget about it.

With EFT it's logical to believe people will follow the same pattern. They'll receive a statement and still believe it's accurate. They'll be relieved of financial management as well as paying bills and the like.

For public relation purposes you can not advertise the vulnerabilities – if there are any.

Along with taking away personal financial control, EFT invites other problems, perhaps the primary one being EFT's potential to infringe on an individual's right to privacy. Said the NCEFT in its report, "The notion of privacy refers to the individual's expectation of control over what information about himself is communicated to or used by others. The object of the consumer's concern regarding privacy under EFT is the potential use of his financial transaction information to develop a personal profile."

In a case concerning government access to bank records, Supreme Court Justice Douglas, in a dissenting opinion, said, "In a sense a person is defined by the checks he writes. By examining them, the agents get to know his doctors, lawyers, creditors, political allies, social connections, religious affiliation, educational interest, the papers and magazines he reads and so on ad infinitum."

NCEFT quoted another court case that stated, "For all practical purposes, the disclosure by individuals or business firms of their financial affairs to a bank is not entirely volitional, since it is impossible to participate in the economic life of contemporary society without maintaining a bank account . . . . Indeed, the totality of bank records provides a current biography."

In their final report to the President and Congress, the NCEFT members said, "Present legal safeguards for the privacy of financial transaction information, irrespective of their sufficiency today, are not adequate to deal with threats to privacy that may arise with EFT."

According to the commission, recent computer science advances increased public awareness of the privacy issue because "this technology permits more efficient data collection, storage and dissemination."

Computer technology itself does not threaten consumers, but the fact that it "could make violations of privacy easier and less expensive to accomplish" does.

As a whole, the NCEFT recognized five specific ways in which EFT "might" jeopardize individual privacy.

First, EFT will generate financial transaction records where none previously existed. Using a debit card instead of cash will create, where there was none before, a record linking a transaction to a particular individual. Although the process already exists when credit cards are used in place of cash, the use of debit cards will just further increase the amount of new financial information collected.

Secondly, EFT will increase the amount of information included in transaction records. While checks record the payor, the payee, the amount and the transaction data, point-of-sale-EFT-debit transactions also include the transaction time and location as well as a "transaction identifier" (whether it was a deposit, cash withdrawal, account transfer or purchase).

Thirdly, in an EFT system finanacial records will be maintained electronically, thus making them easy to retrieve. Today's decentralized (both paper and electronic) personal financial information storage system provides, through access difficulty and high costs, some degree of protection.

Fourthly, EFT systems that operate online and in real time could be used to locate individuals whenever they conduct a transaction.

Finally, EFT may increase the number of institutions with access to an individual's financial records. High start up costs and the need for large transaction volumes for profitability encourage banks to team up to provide EFT services. And many states' laws require competing depository institutions to share EFT systems.

In a shared EFT system many of the participating institutions would maintain records of some sort on each transaction handled. Shared systems also require switches to route messages among participating merchants and depository institutions. Because of these switches, a temporary backup of records called "memo files", which are updated for each transaction, might be necessary for control and audit purposes. In that case, the records kept at the switch, however temporarily, would add another source of stored information.

While collecting and disseminating information on individuals lends itself to potential privacy problems, this information flow helps provide a number of consumer benefits, such as the ability to have checks and credit cards accepted away from home by merchants (who rely on check and credit authorization and guarantee services) or to have loans speedily approved. The difficulty lies in designing measures to protect privacy so potential dangers are forestalled, while not interfering with the benefits that make information flow possible.

Among other legislation, the NCEFT points to the Bank Secrecy Act which requires banks to store financial transaction information for law enforcement purposes. This law increases the "prospect of privacy abuse" because of the types and amount of information it requires depository institutions to collect and store.

According to the NCEFT, "Consumers view the Federal Government as a primary threat to the privacy of individual financial transaction information. If a government agency seeks access to an individual's records from his bank, neither the government nor the bank is under obligation to inform the individual of this action."

Even if a bank wishes to protect its customer's privacy it can only warn the customer that the government is seeking access to his records and then require the government to use formal legal process to obtain records the bank is not already required, by law or regulation, to report.

If you come up with somebody that knows how to detect computer crime, let me know. – FBI Agent Bill Colvin

"I think the potential, the capability, for abuse is certainly made easier (by the use of computerized banking)," said Robert Thompson, administrator of the Operations Planning and Research Staff in the Treasury Department. "I don't mean that it's automatic that people's rights of privacy are going to be invaded. I don't think that way, but I believe the technology certainly lends itself more to that than in the past."

Thompson, who served on the NCEFT along with Treasury's Assistant Fiscal Assistant Secretary, Lester W. Plumly, doesn't believe threats of privacy invasion will impede EFT's development. "We're in the computer era," he said, "so I think these dangers probably are here anyway – whether we have EFT financial transactions or not."

Others believe the privacy issues have been overemphasized; NCEFT member, Ralph F. Lewis, the editor and publisher of the *Harvard Business Review*, said, "There are many places where you can go to a file drawer and pick up everything that you could ever pick up on a tape."

Storing data on tape doesn't make its acquisition any easier, said Lewis, "Particularly if there are appropriate protections as to who can get to the tape and how you get to it.

"Properly designed computer systems will have entrance codes that are foolproof. They'll also be changed every two weeks," Lewis continued. "So, I'm not afraid of some smart technician finding out how much money you or I owe, or what my credit record's been or anything like that."

Eugene Tangney, who represented commercial banks on the NCEFT, basically agreed with Lewis. "Privacy is far more secure under EFT than it is under the present system," he said. "Under the present system we couldn't afford to keep extensive private information about you on our computers; it would take up too much room.

"With EFT, once you came in and got yourself authorized for a line of credit or authorized for a deposit account, the information would be kept internally, none of it online for anyone to look at."

To illustrate his point, Tangney said: "Take a young college girl who wants to cash a check in a store. What does the store want from her to cash the check? Her driver's license – that means she has to give her name and her address – her phone number, and where she works or goes to school.

"Let's talk about privacy. Now, she doesn't know who the clerk behind the counter is. Yet, she's giving up some of the most vital information about her whereabouts just to cash a check," he said. "Compare that with going in with her electronic check cashing card, putting the magnetic strip over the reader, inputting a secret code, getting it approved and then paying.

"And that's the point. You have to think about the present system and then think about EFT. You can't look at one without comparing it to the present system – the one we all fully accept."

The bankers say there's no problem with security," said Walter Anderson, associate director in charge of automated data processing at the Government Accounting Office. "My object is to assure that there's no problem, but I wouldn't say there's no problem.

"I'm worried about security," he continued, "when I said safeguards are adequate I meant I think they will be. In the meantime people are avoiding the kinds of transactions that are subject to interception.

"For public relations purposes you can't advertise the vulnerabilities – if there are any," said Anderson, who represented, along with Donald L. Scantlebury, the GAO on the NCEFT. "If there are any, we'll find them."

But can they find them in time? John M. Carroll, an information scientist who has been an intelligence officer, journalist, and security specialist, wrote in his recent book, *Computer Security*, "There is no computer system in existence at the time of writing that has not been penetrated, if not by thieves, then by 'Tiger Teams' of EDP (electronic data processing) experts who test the effectiveness of protective measures by breaking them in authorized exercises." To date, according to government sources, no Tiger Team has ever failed to penetrate its target computer.

Of course the gap through which a Tiger Team penetrates can be sealed after its demonstration, but what of unauthorized penetrations where the gap is not reported? Undetected system weakenesses cannot be strengthened.

A knowledgeable person attempting to penetrate a supposedly secure computer begins with an advantage – he has to find only one significant flaw in order to start working his way into the system whereas the designers of a system are theoretically afforded no leeway at all in their attempts to make it secure.

After penetrating computer systems, intruders rarely, if ever, leave any trace of their past presence. When gaining control they usually ignore the internal audit trail, built into most systems, that would record their actions and the terminal from which they operated. Then, just before they leave, or sign-off, the system, they merely erase the internal audit system. And no one is the wiser.

Even the Federal Bureau of Investigation recognizes this predicament. But the FBI doesn't attempt to detect computer crimes. "What we do," said special agent Bill Colvin, who trains other agents to investigate computer crimes at the FBI Academy in Quantico, VA, "is try and go in and build a case after a crime has been detected and reported to us." And that's the hard part – detecting a computer crime.

According to Colvin, the FBI's training program is geared to look at "how computer-related crimes have been perpetrated so we can recognize them and recognize where to go for evidence." And he added, "If you come up with somebody that knows how to detect computer crimes, let me know."

Aside from sophisticated breaches of bank security and elaborately planned crimes to rip-off bank computers, EFT system customers face another, more simple, threat to their privacy rights – informal information exchanges.

Even if you have legal rules that say access cannot be obtained, there are still informal exchanges between people. So-and-so knows so-and-so who can get to the computer for this and that information – all you have to do is put it all together.

The extent to which personal privacy is threatened depends on how much government control the banks would receive. Which, while it creates problems in and of itself, might resolve certain issues also.

To achieve additional controls, NCEFT recommended the government limit the amount of EFT system-generated information it requires institutions to maintain and report on individuals and to minimize the extent to which it requires EFT systems to collect information not necessary, as a matter of sound business practice or customer needs, to the system's operation. Additionally, the commission recommended "that EFT systems should not be used for surveillance of individuals, either as to their physical location or patterns of behavior."

Of course, the NCEFT does not ignore the legitimate needs of law enforcement agencies. It suggested the government notify individuals whose records it seeks and give them a chance to contest. Access without prior notice should be allowed only when a court determines advance notice would allow a reasonable possibility of compromising the effectiveness of the law.

To minimize additional threats on privacy, NCEFT recommended EFT institutions treat their customers' records as confidential and only disclose portions of that information when they are essential to the system's operation or when a customer specifically consents to the disclosure.

To help maintain the integrity of an EFT system, NCEFT suggested EFT institutions have the freedom to give information, on their own initiative, to law enforcement officers concerning fraud, attempted fraud, or other crimes involving EFT transactions.

Since financial institutions and other credit grantors (Bank Americard, American Express, restaurants, etc.) will use EFT-system-collected information to make decisions regarding consumers' eligibility for credit or for account services, the consumer should be able to review the accuracy of his own financial records.

While NCEFT recommendations resemble policies established for Federal agencies by the Privacy Act of 1974 and for credit bureau records by the Fair Credit Reporting Act, the extent to which these acts cover various kinds of EFT transactions is unclear. To eliminate ambiguity, particularly in the latter act, the commission suggested broadening it to include the data bases used for EFT transactions and for check and credit authorization services and guarantee services. These modifications, according to the commission, would "enable" consumers whose checks, debit cards or credit card applications are not accepted because of these services to examine the information collected about them, and to correct the data where appropriate.

(Continued on page 66)

National EFT Commission

"It was a fascinating Donnybrook," said Ralph Lewis, a public representative on the National Commission on Electronic Fund Transfers (NCEFT). Composed of 26 people, each of whom held a special interest ("Maybe three of us didn't have an axe to grind," Lewis said), NCEFT produced recommendations for implementing an Electronic Funds Transfer (EFT) system in the United States.

"Everybody had something they wanted to get out of this, which demonstrates the democratic process because we got together on most things," Lewis added.

Established by Public Law 93-495 (Oct. 29, 1974), NCEFT was directed to conduct "a thorough study and investigation of the emerging payment system alternative known as electronic funds transfer and to recommend appropriate administrative action and legislation to permit the orderly development of private and public EFT system," according to its final report.

One government source who followed the commission's development, said there was some dissatisfaction with President Ford because he took more than a year (from NCEFT's creation) to appoint commissioners. And so, the commission, though created in October 1974, did not meet for the first time until February 1976. It turned in its final report on October 28, 1977 – three years to the day after its creation.

"From the outset, the commission was very active," said Walter Anderson, who filled in for the Government Accounting Office's Donald Scantlebury when he became ill. "They (the commissioners) decided to meet monthly and formed four committees in areas of interest and each of those met approximately monthly. In total, there were over 200 meetings, 11 days of hearings and well over 100 witnesses."

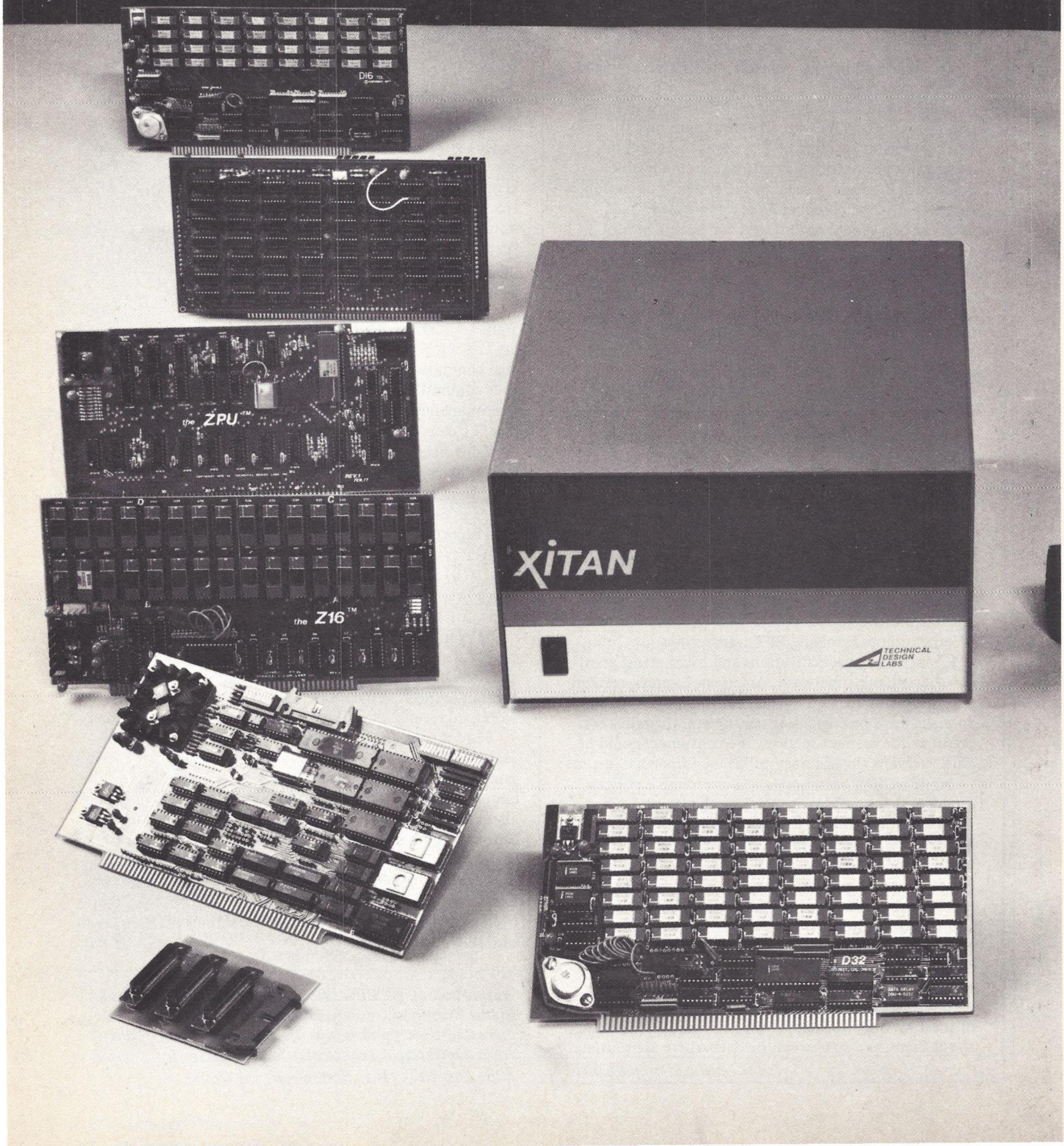
In his letter accompanying the final report, the commission's chairman, William Widnall, a public representative from New Jersey, said the commission testified before Congress on four occasions and presented (in addition to the final report) "two formal documents: a progress report, 'Programs, Plans, Accomplishments of the NCEFT' (Oct. 29, 1976) and an interim report containing the Commission's first substantial body of findings and recommendations, 'EFT and the Public Interest' (Feb. 23, 1977)".

"The interim and final reports don't represent the entire output," said Anderson, with regard to the commission's findings. "There were some 60 working documents – a substantial printed record of the commission's research work and results. Plus, there's a transcript of oral testimony."

Besides four public representatives, some of the representatives on the NCEFT included individuals from these organizations: credit unions, mutual savings banks, state regulators of banks, the Federal Trade Commission, savings and loan associations, commercial banks, the Federal Communications Commission, the Federal Reserve System, the United States Postal Service, the Federal Home Loan Bank Board, organizations providing interchange services for credit cards issued by banks, the General Accounting Office, the Comptroller of the Currency, non-bank institutions offering credit card services, and retailers. *– H.P.B.*

Author's note: NCEFT's final report: EFT in the United States: Policy Recommendations and the Public Interest, *and all of the commission's other documents are available from the National Technical Information Service (NTIS), 5285 Port Royal Rd., Springfield, VA 22161.*

The Ultimate in Microcomputer Hardware & Software!




XITAN
XITAN
XITAN
XITAN
ZAPPLE 12K BASIC
USER'S MANUAL
TDL FORTRAN
IV
USER'S MANUAL
TECHNICAL DESIGN LABS
INSERT
markette
FORTRAN IV
Technical Design Labs
Version 2.1
Princeton, New Jersey 08540
Copyright 1977

In addition to recognizing threats to an individual's privacy and inaccuracies within the system, the NCEFT recognized the possibility and problems associated with debit or check verification card thefts and subsequent unauthorized use.

Once someone breaks the security of a computer, large amounts of money are not any more difficult to steal or embezzle than small ones.

"To protect EFT account holders from incurring any liability when they have not been negligent" and to protect them from "vague negligence standards", the NCEFT recommended the government enact, as Federal legislation, the following four "provisions":
1) EFT account holders should not have to bear any liability for unauthorized use of their cards unless the depository institution can prove, "without benefit of inference or presumption", that account holder negligence contributed to the unauthorized use and that the depository institution exercised "reasonable care" to prevent the loss.
2) "Failure to notify the depository institution of an unauthorized use within 30 days after the receipt of a descriptive statement containing an unauthorized use should render the customer liable for any subsequent unauthorized use that could have been prevented by timely notification."
3) EFT account holders should not bear liability for unauthorized use that takes place after reporting that the device that accesses his account (debit card) has been lost or that the security of his personal identification number (PIN) has been violated.
4) "The customer's liabilities under this set of recommendations should not be increased by contract."

The first recommendation requires the cardholder to exercise "prudence" in the way he handles his debit card and his PIN. Additionally, this first recommendation places a "heavy burden" on any depository institution that seeks to hold the cardholder liable for an unauthorized use of his account.

With the second recommendation, the cardholder must exercise diligence to detect unauthorized use of his account.

The third recommendation parallels the rule for lost or stolen credit cards. An account holder is not responsible for unauthorized transactions once he reports his lost or stolen card.

Finally, the fourth recommendation would allow depository institutions to shoulder a greater risk for unauthorized EFT account use than the guidelines require, but would not allow that institution to disclaim or limit its liability beyond the recommendations.

By placing all liability for all customer fund thefts to the depository institution unless it can prove that, by ignoring the first recommendation's specific warning, the customer contributed to the loss, institutions are compelled to develop and use the most secure technology possible for customer identification.

While recognizing potential threats to individual privacy and the risk of debit card thefts, the NCEFT played down theft from accounts through access to the EFT system itself. Said the commission, "Although such concerns are not entirely without foundation, they are heightened by misunderstandings about EFT and the protection available to account holders. They may also be heightened by popular notions of computer criminals who are capable of penetrating seemingly secure systems or of pranksters intent on disruption of the system."

At the FBI training academy, Bill Colvin said, "I don't know if an EFT environment is going to open up that many more opportunities (for people to steal from bank accounts). It might change some opportunities. By the same token, I think it might make some opportunities that presently exist not as easy."

Because of the limited number of working systems at this early stage of development, there has been only limited experience with criminal penetration of EFT systems. Although there are some points of vulnerability, experience to date shows that actual loss from fraud through EFT systems is low.

In the *New Yorker,* Thomas Whiteside wrote, "The haul from computer crime tends to be very handsome compared with that from other crimes. For example, the average take in armed bank robberies committed in this country, the Federal Bureau of Investigation says, is currently $10,000. The average amount from all reports of missing funds from banks, including bank fraud and embezzlement, according to FBI statistics, is about $19,500. While some reported computer crimes involve the theft or embezzlement of only thousands of dollars, quite a few involve very large sums; a million dollars from a computer crime is considered a respectable, but not an extraordinary, score. Once someone succeeds in breaking the security of a computer system, large amounts of money are not necessarily any more difficult for him to steal or embezzle than small ones."

With the EFT security environment constantly evolving, there is a need to monitor EFT developments closely. Those concerned must be both alert and flexible, meeting the challenge of maintaining effective security in the future.

The commission recommended that state and Federal laws be strengthened to prohibit introduction of fraudulent data into a computer system, unauthorized use of computer related facilities, fraudulent or malicious alteration or destruction of computer data, information or files, and theft, by electronic means or otherwise, of money, financial instruments, property, services, software or data.

Robert Thompson, from the Treasury Department, doesn't believe legislation, in and of itself, will ever necessarily be adequate to guard against privacy problems. "Anyone inclined to misuse or commit any sort of misdeed – whether it be criminal or not – will probably measure the risk against rewards. And I don't know if the existence of legislation and penalties is going to add something to the risk side. I guess legislation and penalties can help, but I wouldn't even begin to say it would stop such things from happening." PC

MUSIC from A to G

BY LINDA M. SCHREIBER

Choir directors, teachers and those who enjoy a good sing-along with friends or family often face songs written out of vocal range of the group. When this problem occurs, there are two choices. The song can be eliminated from the repertoire, or someone can transpose the piece of music to a suitable key. Although the latter is not difficult, it's time consuming and tedious.

"Music Transposer" eliminates the boring task of transposing music to a new key. Having used this method several times, I find the program saves me about one-third the time normally required to rewrite a particular piece of music. This includes time required to copy the transposed piece from print to staff paper. Actual running time of the program is negligible and I often play or sing the piece from the print-out itself.

The program allows the user to input names of the notes from the music sheet into a data line. The user then states the original key of the piece and also the key of the desired transposition. The program then outputs the note names in the new key.

"Music Transposer" was written in

Sample Run 1

"Chorale (Herzlich Thut Mich Verlangen)"

```
THIS PROGRAM WILL TRANSPOSE ANY PIECE OF MUSIC
FROM THE ORIGINAL KEY TO ANOTHER KEY.

ENTER THE NOTES YOU WANT TRANSPOSED INTO
DATA LINES 600-1000 WHEN THIS PROGRAM STOPS.
USE THE LETTER NAMES OF THE NOTES, A '#' FOR SHARP AND
A ''' FOR A FLAT. ENTER AN 'X' AT THE END OF THE PIECE.
BE SURE TO PLACE A COMMA BETWEEN  THE NOTES.

EG. 600 DATA C,C,E,G#,G,X
AFTER ENTERING THE DATA, TYPE THE DIRECT COMMAND-
GOTO 150      -THIS WILL START THE PROGRAM AGAIN.
BREAK IN 140
OK
600 DATA F#,B,A,G,F#,F#,G,A,E,F#,F#,C#,D,D,C#,D,E,A#,B,B
601 DATA F#,B,C,B,A,G,G,F#,E,D,C#,D,E,E,F#,F#,C#,D,E,D,D
602 DATA E,F#,G,C#,D,E,D,C#,B,D,C#,B,A,B,C#,D,D,A,B,A,A
603 DATA G,F#,D,C#,E,D,C#,B,C#,D,D,C#,C#,F#,G,F#,E,A,G,A,F#,X
GOTO 150
WHAT KEY IS THIS PIECE IN? D
WHAT KEY DO YOU WANT TO TRANSPOSE TO? B'
HERE IS YOUR TRANSPOSED SONG:
D   G   F   E'   D   D   E'   F   C   D   D   A   B'   B'   A
B'   C   G'   G   G   D   G   A'   G   F   E'   E'   D   C   B'
  A   B'   C   C   D   D   A   B'   C   B'   B'   C   D   E'   A
   B'   C   B'   A   G   B'   A   G   F   G   A   B'   B'   F
G   F   F   E'   D   B'   A   C   B'   A   G   A   B'   B'   A
 A   D   E'   D   C   F   E'   F   D
```

Altair Extended BASIC and the program runs on an Altair 8800b computer, using between 4K and 5K of memory.

After instructing you how to input notes into the data lines, the program stops and allows you to enter note names of the music as data. The user first types the line number and DATA. You then enter the note names – each note separated by a comma. The last data entry should be an "X".

There is enough space between the last line of the program and the END statement for a long piece of music. When data entries are completed, the user types "GOTO 150" as a direct command. The program now resumes execution at line 150.

Lines 160-200 place the note names located in data lines 390-550 into a 17 x 17 array. The first note in each column of the array is the starting note of that particular key. When you input the key that the music was originally written in, the program scans the first note in each column for that note (lines 210-250). The value of Z is then set to the column number that contains the key of the original music.

In lines 260-300 the user inputs the new key for the transposed music. After finding the column that contains the new key in the array, Y is set to that column number.

The program then executes the routine in lines 310-380. Here the music is read one note at a time from the data lines which were entered by the user. Each note is compared to the names of the notes in the column that is in the original key. When the two note names match (line 350) the program prints the new note in the transposed key (line 370).

After the entire piece of music has been transposed, you can then play directly from the printout. Once the song is entered into the data lines, it can be transposed as many times as desired.

The first program run shows the transposition of the composition "Chorale (Herzlich Thut Mich Verlangen)" by J.S. Bach from D to B′. The second run transposes the guitar chords for the old English ballad "Greensleeves" from G to E.

If your BASIC doesn't allow direct commands within a program, you can modify this program by eliminating line 140 and entering the notes to be transposed into data line 600-1000.

Sample Run 2

Greensleeves

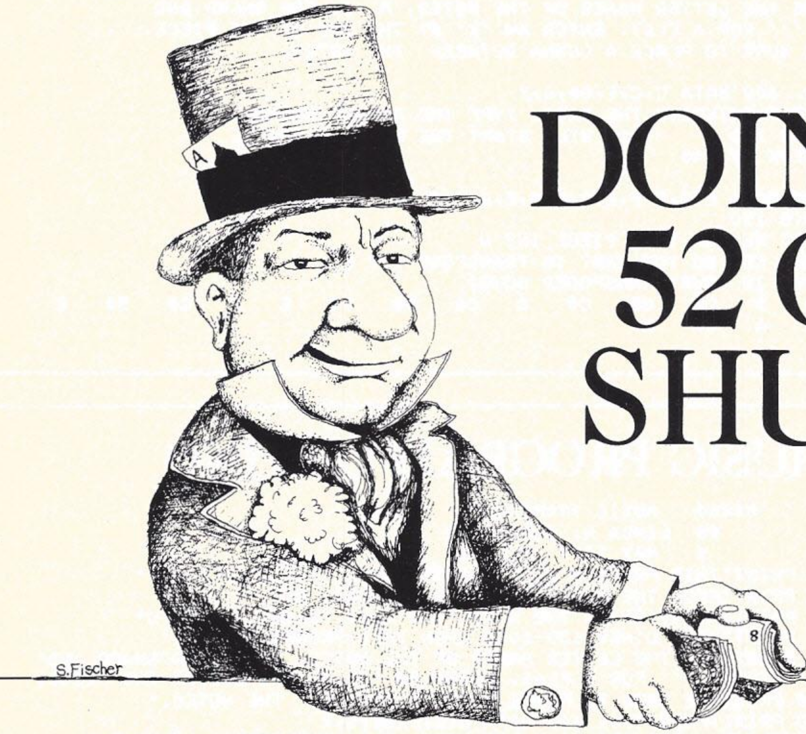
```
ENTER THE NOTES YOU WANT TRANSPOSED INTO
DATA LINES 600-1000 WHEN THIS PROGRAM STOPS.
USE THE LETTER NAMES OF THE NOTES, A '#' FOR SHARP AND
A ''' FOR A FLAT. ENTER AN 'X' AT THE END OF THE PIECE.
BE SURE TO PLACE A COMMA BETWEEN  THE NOTES.

EG. 600 DATA C,C,E,G#,G,X
AFTER ENTERING THE DATA, TYPE THE DIRECT COMMAND-
GOTO 150      -THIS WILL START THE PROGRAM AGAIN.
BREAK IN 140
OK
600 DATA E,D,E,B,E,D,E,B,E,G,D,E,B,G,D,C,B,E,X
GOTO 150
WHAT KEY IS THIS PIECE IN? G
WHAT KEY DO YOU WANT TO TRANSPOSE TO? E
HERE IS YOUR TRANSPOSED SONG:
C#   B   C#   G#   C#   B   C#   G#   C#   E   B   C#   G#   E
 B   A   G#   C#
```

MUSIC PROGRAM LISTING

```
10 '  *****   MUSIC TRANSPOSER   *****
20 '     **  LINDA M. SCHREIBER  **
30 '      *   MAY 1977  REV. 2   *
40 PRINT"THIS PROGRAM WILL TRANSPOSE ANY PIECE OF MUSIC"
50 PRINT"FROM THE ORIGINAL KEY TO ANOTHER KEY."
60 PRINT:PRINT"ENTER THE NOTES YOU WANT TRANSPOSED INTO"
70 PRINT"DATA LINES 600-1000 WHEN THIS PROGRAM STOPS."
80 PRINT"USE THE LETTER NAMES OF THE NOTES, A '#' FOR SHARP AND"
90 PRINT"A ''' FOR A FLAT. ENTER AN 'X' AT THE END OF THE PIECE.
100 PRINT"BE SURE TO PLACE A COMMA BETWEEN  THE NOTES."
110 PRINT:PRINT"EG. 600 DATA C,C,E,G#,G,X
120 PRINT"AFTER ENTERING THE DATA, TYPE THE DIRECT COMMAND-"
130 PRINT"GOTO 150      -THIS WILL START THE PROGRAM AGAIN."
140 STOP
150 DIM K$(17,17)
160 FOR I=1 TO 17 'PLACE THE NOTES IN AN ARRAY
170 FOR J=1 TO 17
180 READ K$(I,J)
190 NEXT J
200 NEXT I
210 PRINT"WHAT KEY IS THIS PIECE IN";
220 INPUT Z$
230 FOR Z=1 TO 17 'FIND THE KEY THE MUSIC IS WRITTEN IN
240 IF Z$=K$(1,Z) THEN 260
250 NEXT Z
260 PRINT"WHAT KEY DO YOU WANT TO TRANSPOSE TO";
270 INPUT Y$
280 FOR Y=1 TO 17 'FIND THE NEW KEY
290 IF Y$=K$(1,Y) THEN 310
300 NEXT Y
310 PRINT"HERE IS YOUR TRANSPOSED SONG:"
320 READ S$
330 IF S$="X" THEN 1001
340 FOR N=1 TO 17
350 IF S$=K$(N,Z) THEN 370 'FIND THE NOTE IN THE ORIGINAL KEY
360 NEXT N
370 PRINT K$(N,Y)"    "; 'PRINT IT IN THE NEW KEY
380 GOTO 320
390 DATA C,C#,D',D,D#,E',E,F,F#,G',G,G#,A',A,A#,B',B
400 DATA C#,D,D,D#,E,E,F,G',G,G,G#,A,A,A#,B,B,C
410 DATA D',D,D,D#,E,E,F,G',G,G,G#,A,A,A#,B,B,C
420 DATA D,D#,E',E,F,F,F#,G,G#,A',A,A#,B',B,C,C,C#
430 DATA D#,E,E,F,F#,G',G,A',A,A,A#,B,B,C,C#,D',D
440 DATA E',E,E,F,F#,G',G,A',A,A,A#,B,B,C,C#,D',D
450 DATA E,F,F,F#,G,G,G#,A,A#,B',B,C,C,C#,D,D,D#
460 DATA F,F#,G',G,G#,A',A,B',B,B,C,C#,D',D,D#,E',E
470 DATA F#,G,G,G#,A,A,A#,B,C,C,C#,D,D,D#,E,E,F
480 DATA G',G,G,G#,A,A,A#,B,C,C,C#,D,D,D#,E,E,F
490 DATA G,G#,A',A,A#,B',B,C,C#,D',D,D#,E',E,F,F,F#
500 DATA G#,A,A,A#,B,B,C,D',D,D,D#,E,E,F,F#,G',G
510 DATA A',A,A,A#,B,B,C,D',D,D,D#,E,E,F,F#,G',G
520 DATA A,A#,B',B,C,C,C#,D,D#,E',E,F,F,F#,G,G,G#
530 DATA A#,B,B,C,C#,D',D,E',E,E,F,F#,G',G,G#,A',A
540 DATA B',B,B,C,C#,D',D,E',E,E,F,F#,G',G,G#,A',A
550 DATA B,C,C,C#,D,D,D#,E,F,F,F#,G,G,G#,A,A,A#
1001 END
```

If you want to make a card shark out of your poker-faced computer, you must first teach it to shuffle. Here's a simple, elegant routine that lets your micro handle a deck like a riverboat gambler . . .



DOING THE 52 CARD SHUFFLE

BY ANDREW RUSSAKOFF

Many computer games, like card games, demand the computer begin with a random ordering of fixed elements (for example, cards). This shuffling can be done using random numbers, but needs some good bookkeeping. Before attempting to shuffle a deck of cards, consider a simpler example.

Games like Mastermind and Comp IV require the computer to produce four items which the player must guess in a series of steps. Each guess produces an ambiguous response indicating how many items you guessed correctly, but not which ones. The computer uses its ability to produce random number to select the four target items.

We can write statements that say choice one (call it C(1)) is not equal to C(2), and so forth; but a series of IF/THEN tests gets very long even for a small ordering. So let's set up a bookkeeping list, B( ), where all the original numbers have value B( )=0. When any number, say 3, is chosen, let B(3)=1. If later the computer offers 3 as the next choice, the B(3) value indicates it must choose again.

Now let's extend this model to the larger problem of producing a random ordering of 52 numbers – a shuffled deck of cards. As a reminder, the conversion of a number to a card follows from the identification of the suit (clubs=0, diamonds=1, hearts=2, and spades=3), S=INT ((R-1)/13), and the actual value of the card (1=Ace, 2=Two, 3=Three, etc.), V=R-13*S.

Select a random number as the first number (card), another as the second, again keeping one list A( ) for the random ordering and another list for the "deck" from which we drew them, B( ). This B( ) list tells us if the number has already been chosen.

The beauty of this record-keeping is its clarity and efficiency. No need to compare our current candidate with the entire deck we've already selected. Instead, we just refer to the B( ) list.

Still, the last few choices come slowly. Newly selected random numbers will most often lie among the numbers already used, wasting lots of machine time. We can reshape the problem after picking half the deck, but this procedure is very involved. A much slicker, faster, more elegant shuffling exists, using only one list, A( ).

Start with all the cards in order, that is, A(1)=1, A(2)=2, etc. Then for each random number picked, say 15, we switch items A(1) and A(15). This interchanging is not effective; since the computer chooses only 52 random numbers, many cards may be left in place.

A more convincing shuffle along the same lines counts the number of random numbers produced. For example, if the first step produces random number 15, the computer interchanges places A(1) and A(15). Then, if the second step produces random number 32, the computer interchanges places A(2) and A(32). This last variation produces a shuffled deck with precisely 52 random numbers, since we move each A(I) at Step I. (See Program Listing.)

But we must be careful of the switch itself. If A(3)=8 and A(15)=10, we cannot command A(3)=A(15), then A(15)=A(3). This technique would result in both variables having value 10. We need to hold onto the "8", so let H=A(3); then A(3)=A(15); and finally, A(15)=H.

Just make sure you don't deal from the bottom of the deck!

PROGRAM LISTING

```
 10 DIM A(52)
 20 FOR N=1 TO 52
 30    LET A(N)=N
 40 NEXT N
 50 FOR K=1 TO 52
 60    LET R=INT(52*RND(5)+1)
 70    LET H=A(K)
 80    LET A(K)=A(R)
 90    LET A(R)=H
100 NEXT K
110 END
```

Illustration by Stephen Fischer

Playing the Game

Game Playing with BASIC
by Donald G. Spencer, $6.95

If you ever develop an occasional desire to get between the covers with something warm, entertaining and mysterious – then this book is for you. Depending on the type of player you are, you should get many moments of satisfaction from the 160 pages of games, puzzles and other mysteries you can play in BASIC on your microcomputer.

Actually, the book is more than a motley collection of games and programs. First of all you are introduced to the computer – a social grace often overlooked by computer stores. Following the introduction the book unfolds a short, entertaining history of game playing.

"Games have been with us since the dawn of man," says author Donald Spencer. "Ancient Chinese and Egyptians devised games such as Nim, Awari, Tower of Hanoi, Fan Tan and Go. Who in the United States does not know how to play Tic-Tac-Toe, Checkers, or Monopoly? Who does not, now and then, indulge in playing with solvable puzzles of one kind or another?

"Games and mathematical recreations share three characteristics: First, they are logical; second, they are fun, third, they appear to be quite useless. Why should anyone indulge in anything that is of no practical value?

"Prime numbers, for example, have no practical value. It may be decades, if ever, before they can be put to use. However, the study of prime numbers has filled many gaps in the field of *number theory* – that mathematical discipline which studies the basic properties of all numbers. The study of the patterns of certain types of Magic Squares has reduced the number of experiments required to obtain plant growth and radiation data. Many games still seem to be useless, but who can say what next year or the next decade will bring?

"Until the invention of the digital computer, game playing was limited to human beings. Today, students, teachers, programmers, analysts and game playing novices spend considerable time programming computers to play games. The educational value of playing and writing computer games is substantial. They offer ideal supplemental learning experiences in problem solving, probability, computer programming, statistics, logic and decision making."

Fibonacci Numbers

A man bought a pair of rabbits and bred them. The pair produced one pair of young after one month, and a second pair after the second month. They then stopped breeding. Each new pair also produced two more pairs in the same way and then stopped breeding. How many new pairs of rabbits did the man get each month?

To answer this question, let us write down in a line the number of pairs in each generation. First write the number 1 for the single pair he started with. Next we write the number 1 for the pair they produced after a month.

The next month both pairs had young, so the next number is 2. We now have three numbers in a line: 1, 1, 2. Each number represents a new generation. Now the first generation stopped producing. The second generation (1 pair) produced 1 pair. The third generation (2 pairs) produced 2 pairs. So the next number we write is 1 + 2, or 3. Now the second generation stopped producing. The third generation (2 pairs) produced 2 pairs. The fourth generation (3 pairs) produced 34 pairs. So the next number we write is 2 + 3, or 5.

Each month, only the last two generations produce, so we can get the next number and all succeeding numbers by adding the last two numbers in the line (see Fig 1). The numbers we get in this way are called *Fibonacci numbers*. The first twelve of them are: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144. They have interesting properties and keep popping up in many places in nature.

(Continued on following page)

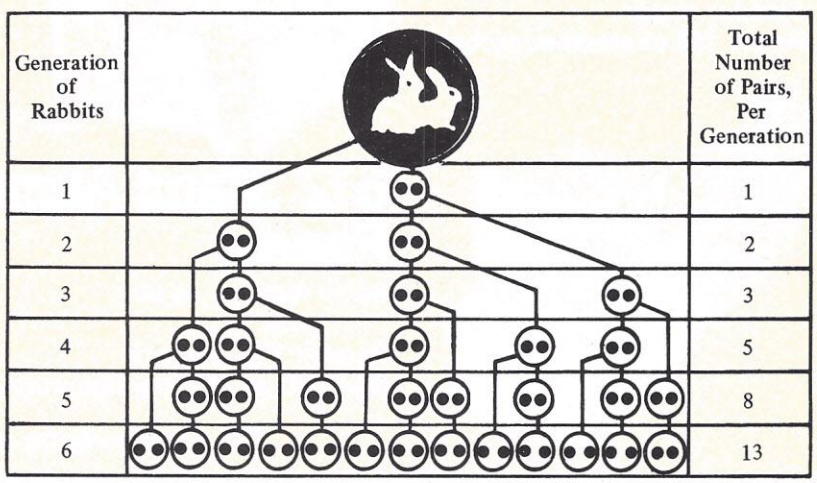


Fig 1 Rabbits and generations

The author describes techniques of developing game programs, gives a brief primer on BASIC and then jumps right into its main theme: games and puzzles. An example of "having fun with numbers" (Fibonacci Numbers) is reprinted here to show what you'll be getting for your money. ▮◖

Fibonacci Numbers

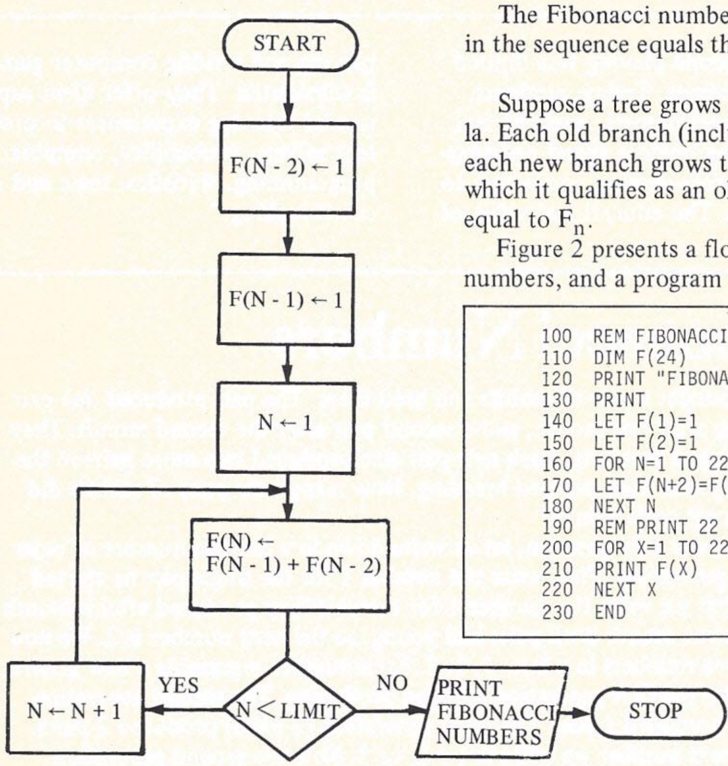


Figure 2 Flowchart for generating Fibonacci numbers.

The Fibonacci numbers are such that, after the first two, every number in the sequence equals the sum of the two previous numbers:

$$F_n = F_{n-1} + F_{n-2}$$

Suppose a tree grows according to the following, not unrealistic, formula. Each old branch (including the trunk) puts out one new branch per year, each new branch grows through the next year without branching, after which it qualifies as an old branch. The number of branches after n years is equal to F_n.

Figure 2 presents a flowchart of a procedure that will compute Fibonacci numbers, and a program that computes and prints 22 of these numbers.

```
100 REM FIBONACCI NUMBERS
110 DIM F(24)
120 PRINT "FIBONACCI NUMBERS"
130 PRINT
140 LET F(1)=1
150 LET F(2)=1
160 FOR N=1 TO 22
170 LET F(N+2)=F(N+1)+F(N)
180 NEXT N
190 REM PRINT 22 FIBONACCI NUMBERS
200 FOR X=1 TO 22
210 PRINT F(X)
220 NEXT X
230 END
```

```
FIBONACCI NUMBERS

1     144
1     233
2     377
3     610
5     987
8     1597
13    2584
21    4181
34    6765
55    10946
89    17711
```

Reprinted with permission. "Game Playing with BASIC," Donald G. Spencer. Hayden Book Co., Rochelle Park, NJ. Copyright © 1977. $6.95.

Math For Minors

BY MIKE DONAHUE

If you have children in your family or work with youngsters, this may be a practical program for you. Written in SWTPC 4K BASIC, it's essentially a math drill program designed to help children learn mathematical fundamentals. Although the program isn't just for elementary school children, it provides drills using all four basic math functions on numbers up to 4 digits.

It's more fun than flash cards, and you can run through the problems by yourself, receiving instant feedback to your answers. Simply, the program gives you a variety of math problems, both easy and difficult.

You choose problem types from a list included as optional program instructions, printed when you start the program. In addition to the four standard math functions (add, subtract, multiply and divide), the selection includes combinations of several logical math functions. For example, addition and subtraction problems may be presented together, randomly. The other combinations are multiply and divide together and a random presentation of all four.

In order to tailor the program to a

Program Listing

```
0010 REM MATH PROGRAM FOR ADD, SUBTRACT, MULTIPLY AND DIVIDE
0020 PRINT "******** MATH TEST ********"
0050 LET F=0
0060 LET G=0
0070 LET E1=0
0080 LET G1=0
0082 PRINT "ENTER 1 FOR INSTRUCTIONS";
0084 INPUT I
0086 IF I<>1 GOTO 170
0090 PRINT "AVALIABLE PROBLEMS ARE:"
0100 PRINT "    1 FOR ADDITION ONLY"
0110 PRINT "    2 FOR SUBTRACTION ONLY"
0120 PRINT "    3 FOR MULTIPLICATION ONLY"
0130 PRINT "    4 FOR DIVISION ONLY"
0140 PRINT "    5- FOR COMBINATION OF ADDITION&SUBTRACTION"
0150 PRINT "    6 FOR COMBINATION OF MULTIPLICATION&DIVISION"
0160 PRINT "    7 FOR COMBINATION OF ALL 4"
0170 PRINT "ENTER YOUR CHOICE";
0175 INPUT P
0180 IF P<1 GOTO 170
0190 IF P>7 GOTO 170
0200 PRINT "ENTER: PROBLEM LIMIT";
0210 INPUT L
0220 IF L<1 GOTO 200
0230 LET A=INT(L*RND(0)+1)
0240 LET B=INT(L*RND(0)+1)
0250 IF G+E =10 GOSUB 1500
0260 ON P GOTO 500,600,700,800,400,450,300
0300 REM RANDOMIZE FOR COMBINATION OF ALL 4
0310 LET S=INT(4*RND(0)+1)
0320 ON S GOTO 500,600,700,800
0400 REM RANDOMIZE FOR COMBINATION OF ADD&SUB
0410 LET S=INT(2*RND(0)+1)
0420 ON S GOTO 500,600
0450 REM RANDOMIZE FOR COMBINATION OF MULT&DIV
0460 LET S=INT(2*RND(0)+1)
0470 ON S GOTO 700,800
0500 REM ADDITION ROUTINE
0510 PRINT A;"+ ";B;"= ";
0520 INPUT C
0530 IF C=A+B GOTO 1000
0540 GOSUB 1200
0550 GOTO 510
0600 REM SUBTRACTION ROUTINE
0610 IF A>B GOTO 660
0620 LET C=A
0630 LET A=B
0640 LET B=C
0660 PRINT A;"- ";B;"= ";
0670 INPUT C
0680 IF C=A-B GOTO 1000
0685 GOSUB 1200
0690 GOTO 660
0700 REM MULTIPLICATION ROUTINE
0710 PRINT A;"X ";B;"= ";
0720 INPUT C
0730 IF C=A*B GOTO 1000
0740 GOSUB 1200
0750 GOTO 710
0800 REM DIVISION ROUTINE
0810 LET A=A*B
0820 PRINT A;"/ ";B;"= ";
0830 INPUT C
0840 IF C=A/B GOTO 1000
0850 GOSUB 1200
0860 GOTO 820
1000 REM CORRECT RESPONSE ROUTINE
1010 LET G=G+1
1020 LET R=INT(3*RND(0)+1)
1030 ON R GOTO 1080,1100,1120
1080 PRINT "VERY GOOD..  ";
1090 GOTO 230
1100 PRINT "EXCELLENT..  ";
1110 GOTO 230
1120 PRINT "CORRECT....  ";
1130 GOTO 230
1200 REM ERROR RESPONSE ROUTINE
1210 LET E=E+1
1220 LET R=INT(2*RND(0)+1)
1230 ON R GOTO 1280,1320
1280 PRINT "* A MISS-TRY AGAIN..  ";
1290 GOTO 1400
1320 PRINT "* YOU MISSED-RETRY..  ";
1400 REM CK TO SEE IF SCORE NEEDED
1410 IF G+E <>10 RETURN
1420 GOSUB 1500
1430 PRINT "DON'T FORGET THIS ONE!  ";
1440 RETURN
1500 REM ROUTINE TO PRINT SCORE
1510 LET E1=E1+E
1520 LET G1=G1+G
1530 LET E=0
1540 LET G=0
1550 LET T=G1+E1
1555 PRINT
1560 PRINT "YOUR SCORE AT THE END OF ";T;"PROBLEMS IS ";
1570 PRINT G1;"CORRECT AND ";E1;"WRONG."
1575 IF T<50 RETURN
1700 REM STOP ROUTINE
1710 PRINT "THANK YOU FOR THE GAME!"
9999 END
```

specific child's math level the problem complexity is variable and controlled by the number you enter as "problem limit". The larger the number you enter, the more difficult the problems and vice versa. For example, by entering "5" the numbers used in the problems will be limited to a range of 1 to 5. This limit holds true for all of the math functions except division, where, for simplicity's sake, the division problems are designed to not have remainders. (The method chosen to insure this was to multiply one problem number by the other.) Thus, the upper limit of the division problems will be higher than the number entered at "problem limit".

The program uses random numbers liberally for math problems, for combinations of problem types and to provide a variety of correct or error response messages.

If you modify this program for other BASICs, check random number use carefully.

Your score is printed after each set of 10 problems. At the end of 50 problems, the game automatically ends. PC

Program Run

```
#RUN
******** MATH TEST ********
ENTER 1 FOR INSTRUCTIONS? 1
AVALIABLE PROBLEMS ARE:
   1 FOR ADDITION ONLY
   2 FOR SUBTRACTION ONLY
   3 FOR MULTIPLICATION ONLY
   4 FOR DIVISION ONLY
   5 FOR COMBINATION OF ADDITION&SUBTRACTION
   6 FOR COMBINATION OF MULTIPLICATION&DIVISION
   7 FOR COMBINATION OF ALL 4
ENTER YOUR CHOICE? 7
ENTER: PROBLEM LIMIT? 10
7 - 6 = ? 1
EXCELLENT..  2 + 3 = ? 5
CORRECT....  3 + 1 = ? 4
CORRECT....  6 + 8 = ? 14
EXCELLENT..  3 X 10 = ? 30
VERY GOOD..  7 - 3 = ? 4
EXCELLENT..  24 / 3 = ? 8
CORRECT....  9 + 8 = ? 17
EXCELLENT..  1 + 3 = ? 4
EXCELLENT..  6 + 9 = ? 3
* A MISS-TRY AGAIN..
YOUR SCORE AT THE END OF 10 PROBLEMS IS 9 CORRECT
AND 1 WRONG.
DON'T FORGET THIS ONE!  6 + 9 = ? 15
EXCELLENT..  1 X 3 = ? 1
* YOU MISSED-RETRY..  1 X 3 = ? 3
CORRECT....  1 + 9 = ? 10
VERY GOOD..  9 + 1 = ? 10
VERY GOOD..  3 X 7 = ? 21
EXCELLENT..  10 + 7 = ? 17
CORRECT....  6 + 9 = ? 15
VERY GOOD..  18 / 9 = ? 2
EXCELLENT..  9 + 8 = ? 17
VERY GOOD..
YOUR SCORE AT THE END OF 20 PROBLEMS IS 18 CORRECT
AND 2 WRONG.
9 - 9 = ? 9
* A MISS-TRY AGAIN..  9 - 9 = ? 0
VERY GOOD..  5 + 7 = ? 12
VERY GOOD..  9 + 8 = ? 17
EXCELLENT..  7 - 5 = ? 2
VERY GOOD..  1 + 2 = ? 3
CORRECT....  7 + 10 = ? 17
CORRECT....  8 - 6 = ? 2
VERY GOOD..  1 + 6 = ? 7
CORRECT....  5 - 1 = ? 4
EXCELLENT..
YOUR SCORE AT THE END OF 30 PROBLEMS IS 27 CORRECT
AND 3 WRONG.
6 - 1 = ? 5
EXCELLENT..  3 X 2 = ? 5
* A MISS-TRY AGAIN..  3 X 2 = ? 6
CORRECT....  5 X 6 = ? 30
CORRECT....  28 / 4 = ? 7
EXCELLENT..  4 / 1 = ? 4
VERY GOOD..  9 - 2 = ? 7
CORRECT....  7 X 3 = ? 21
CORRECT....  6 - 3 = ? 3
VERY GOOD..  10 X 1 = ? 10
EXCELLENT..
YOUR SCORE AT THE END OF 40 PROBLEMS IS 36 CORRECT
AND 4 WRONG.
63 / 9 = ? 7
EXCELLENT..  5 + 7 = ? 12
VERY GOOD..  1 + 8 = ? 9
CORRECT....  7 X 2 = ? 9
* YOU MISSED-RETRY..  7 X 2 = ? 14
VERY GOOD..  3 X 6 = ? 18
VERY GOOD..  16 / 8 = ? 2
CORRECT....  6 - 6 = ? 0
VERY GOOD..  7 - 1 = ? 6
VERY GOOD..  27 / 9 = ? 3
CORRECT....
YOUR SCORE AT THE END OF 50 PROBLEMS IS 45 CORRECT
AND 5 WRONG.
THANK YOU FOR THE GAME!
```

Illustration by Barbara Leonard

The Elf II

A Small Beginning

BY HOWARD G. DRAKE

I took the plunge! I have my very own computer, right here in my own house. Well, maybe it is more like a baby step than a plunge. Right now, my computer has just 256 bytes of RAM, and only understands machine language. But I'm ahead of my story. Let's start at the beginning.

My beginning was reading the Altair story three years ago – I started saving my money, except that money which went for charter subscriptions to *Personal Computing, Byte,* and four or five other computer magazines. I read anything remotely connected to hobby computers; and learned many fascinating and useful facts. So I recommend: Read! Read! Read!

My money started to accumulate. As hobby computers proliferated, they seemed to be getting better. But I kept reading about a glitch in this machine's bus or a bug in that machine's BASIC. I decided to wait for the "next generation".

Well, the next generation, Commodore and Radio Shack, were a quantum jump ahead of previous models in performance per initial investment. But these two machines appeared to be difficult to expand. So, again, I decided to wait. The 16-bit microprocessors were appearing. And if $600 buys an 8-bit PET this year, then next year, $800 will buy a 16-bit Mistress.

But I was anxious to get some hands-on experience. I started thinking about taking some of my savings and buying a KIM. Then I read an article on the $99 Elf II microcomputer from Netronics, (333 Litchfield Rd., New Milford, CT). This was what I wanted. I'd learn how to use a microcomputer, and only make a small dent in my savings account. I could start building and debugging some peripherals in anticipation of my 16-bit "mainframe" that would follow in a couple years.

Photographs by Howard G. Drake

I mailed out my order for the Elf II. Three weeks to the day later, I received a notice from UPS there was a package for me at the local terminal.

At the time I had read the article about Elf II, I saw another ad for Hitachi TVs on sale. I remembered reading somewhere that the Hitachi has a wide bandwidth and an isolation transformer. Also, a kit is out to convert their black and white TV to a video monitor. I rushed out and purchased Hitachi model P-40 for $79. I then scooted home and ordered the TV monitor conversion kit from Pickles & Trout, (P.O. Box 2270, Goleta, CA), $20. The Pickles & Trout kit, TVM 41, arrived two days after the Elf.

And that was the beginning.

The Elf II is a single board microcomputer. It is built around the RCA CDP1802 D. (There is also a CDP1802 CD, but this has a tighter tolerance on supply voltage.)

The 1802 is an 8-bit microprocessor using CMOS. By using CMOS, the unit draws very little current. The 1802 has sixteen general purpose 16-bit registers. It also has DMA (Direct Memory Access), which is very useful in video graphics applications. The 1802 can address up to 65K bytes of memory. Any of the 1802's sixteen general purpose registers can be used for addresses or data. And any of these sixteen registers can be used as the program counter, which is useful for subroutines.

Also, the 1802 has "load and advance" and "store and decrement" commands, which are needed for implementing stacks.

The Elf II also has the RCA CDP1861 CD video display controller. This device works directly with the DMA capabilities of the 1802 to map any segment of memory onto a black and white monitor or modified TV. The resolution is 64 segments horizontally by 128 segments vertically – good for graphics and for large block letters.

Although the included 256 bytes of memory may sound small, they go a long way when using machine language. On the Elf board is a bus with slots for five expansion boards. Netronics said 4K memory boards will be available shortly. And 1802 Tiny BASIC is available from Itty Bitty Computers.

Mounted on the board is a hex key pad for input. The 1802 machine

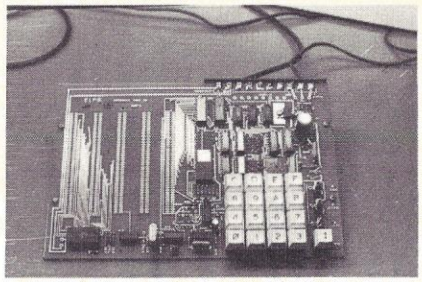


Fig 1. The assembled ELF II. Two Hex digit LED output displays information in each memory location.

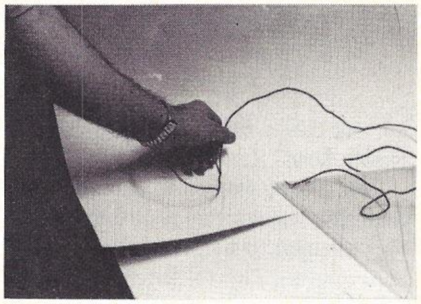


Fig 2. The inexpensive wrist band means the author and the workbench cover are at the same potential. This eliminates chance of static discharge.

language instructions are 2 hex digits long. Output is by a 2-digit, 7-segment display. Also, the Elf board has switches for Input, Memory Protect, Load, and Run. A regulated power supply and a crystal controlled clock are also on the board.

I began construction by setting up an anti-static work bench. Ideas on static control came from a video tape called "Zap! Static Awareness" by Dan Anderson of Richmond Corporation, an excellent presentation on static electricity. I covered the top of my work bench with a 60 mil sheet of anti-static pink polyethylene. A 36" x 60" x 60 mil sheet of pink polyethylene costs about $10.

I punched a hole in the corner of the pink poly sheet, and twisted one end of a five foot stranded wire through the hole. The other end of the wire I attached to an old metal I.D. bracelet from my teeny-bopper days. Whenever I sat at the work bench, I'd slip on the bracelet. I also wore all-cotton shirts with short sleeves, (polyester shirts are loaded with static). I used sockets for all my I.C.s, and, therefore, didn't have to worry about static discharge from my soldering pencil.

Netronics supplies only four I.C. sockets with the kit. The remaining thirteen sockets cost about $6. All are easy to find except the 20 pin socket, which I finally located at Active Electronics Sales Corporation, (P.O. Box 1035, Framingham, MA).

Construction went smoothly and quickly. When I connected the Elf II to the 6.3 VAC transformer (purchased from Netronics for $4.95) everything checked out.

Next step was to convert the TV set to a video monitor, using the Pickles & Trout TVM 41 kit. The TVM 41 allows the Hitachi set to function as either a regular TV or a monitor simply by flipping a switch.

Construction and installation of the TVM 41 was slow, as the Hitachi set was only a month old and I was proceeding cautiously. At one point, the instructions say to test the completed TVM PC board with a 9 volt battery; the output should be 3.5 volts. I tested my board, and got 2.9 volts. I called the company, who informed me it was an error in the instructions; 2.9 was correct.

I then installed the TVM 41 in the TV, and ran the coaxial cables from the TVM 41 to the Hitachi PC board. I turned the set on, and tuned in a local station. I got a great sound, but no picture. After about an hour of checking, I noticed that the braided shield on one of the coaxial cables was shorting out to a switch terminal. After wrapping some black tape around the braided shield, I got a great picture.

I then connected the Elf and the Hitachi. I got an image, but it was very distorted. I changed the sync resistor R34 on the Elf from 10K ohms to 1800 ohms. Also, the input resistors on the TVM 41 board should be 56 ohms and 22 ohms. This resulted in a very clear picture.

Finally, I had a working computer right in my own home. So far, I've displayed the Enterprise moving across the screen, and played a number guessing game.

I'm in the process now of learning 1802 machine language. The 1802's architecture and machine language are explained well in the user manual number MPM-201A (put out by RCA).

One of the output lines of the 1802 can control a light or drive a loudspeaker. As soon as I complete this article, I'm going to master machine language and hook up a loudspeaker. Then look out punk rock; here comes RAM rock!

Game Playing

No Strings Attached

BY BRUCE A. SCOTT

Steve is a nice guy with a small problem you might share. He's one of the lucky few who owns a Z-80-based microcomputer. But that isn't his problem.

Far from it. The Z-80 is a nice piece of equipment with a lot of useful bells and whistles like cassette recorder, built in BASIC, detached keyboard and 12-inch display monitor.

The problem is the limited number of games that will run on Steve's interpreter. His BASIC only has one 1-dimensional array and it only recognizes two string variables, A$ and B$. There aren't many games that fit these limitations. This one does.

I offer it because you may share Steve's problem of limited BASIC and small memory. But you'll find it a lot of fun to play even if you run on a full blown mini with disk and tape backup like I do.

The principle of the game is simple. The computer selects up to four of the first six digits (1 through 6) and arranges them randomly into a secret four-digit number. You type in guesses about the number and win when you guess all digits in the right order. Along the way, the computer gives you hints to guide your guesses until you type in the secret number. (Yes, it is somewhat like Mastermind and Comp IV, but the program is simpler.)

Every digit in your guess has value and position. If your guess is 3456, the first digit has a value of 3 and a position of 1. The 6 has a value of 6 and a position of 4. This information is helpful in understanding the computer's hints.

The computer's hints are made up of "X"s, "I"s, and "-"s. An "X" means that one of the digits in your guess has the same value and position as a digit in the computer's number. Each "I" means that one of the digits in your guess has the same value as one of the computer's digits but the positions are different. The "-" means that one of your digits is not in the computer's number.

Let me give you an example. Suppose that you guess 1234 and the computer responds XXII. Interpret the response to mean that two of your digits are correct in value and position. The other two are correct in value only. The computer's number could be 2134 but there are five other possibilities: 1243, 1432, 1324, 4231 and 3214. You need additional guesses and hints to find which of the possibilities is correct.

Hints can be misleading, so let me give you another example. Suppose you guess 5555 and the computer responds XIII. There would have to be one 5 and three digits that are not 5 in the computer's secret number. The "X" tells you that one of your 5s is in the correct position. The three "I"s tell you that the other three 5s are in the wrong position.

Now suppose that the computer selects a new number. You guess 1233 and the computer responds II - -. What do you know about the secret number? Not much. There could be one, two, or zero 3s in the secret number.

Your guesses must be selected carefully and the hints interpreted correctly before you can hope to find the secret number in a minimum number of guesses.

The remarks in the Program Listing should help you see how it works. (For my system, all remarks start with "!".) A typical game is shown in the Sample Run.

Program Listing

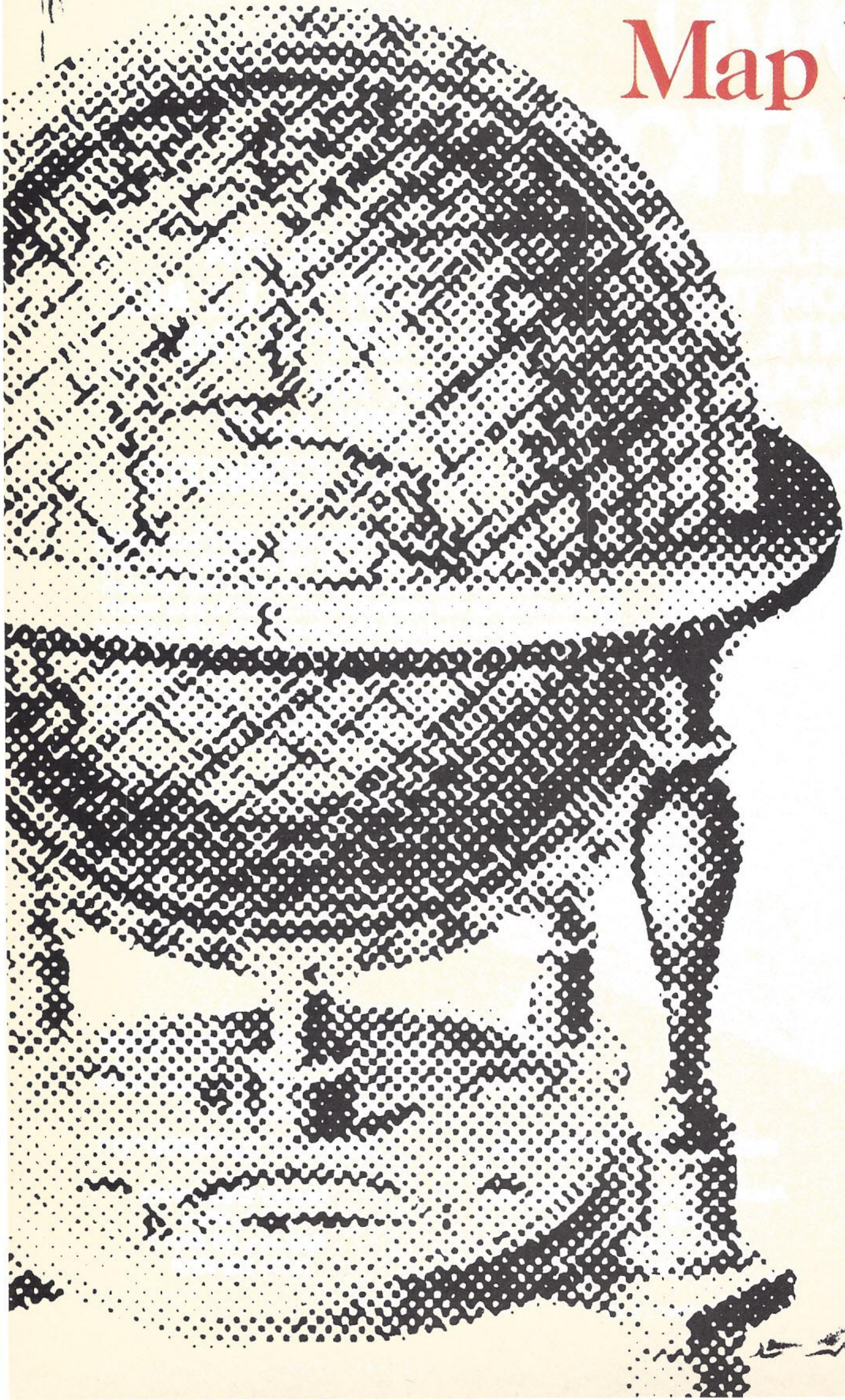
```
120 !FØRM THE HIDDEN NUMBER
130 FØRI=5TØ8:A(I)=INT(RND(0)*6+1):NEXTI
150 !ACCEPT A GUESS
155 N=0
160 N=N+1:PRINT:M=0:INPUTB:B=B/1000:FØRI=1TØ4:
      A(I)=INT(B):B=(B-A(I))*10
170 IFI=3THENB=B+.0005
180 !LØØK FØR MATCHES
190 NEXTI:FØRI=1TØ4:J=I+4:IFA(I)<>A(J)THEN210
200 PRINT'X';:M=M+1:A(I)=17
210 NEXTI:IFM=4THEN320
220 !LØØK FØR SIMILARITIES
230 FØRI=1TØ4:FØRJ=5TØ8:IFA(I)<>A(J)THEN260
240 IFJ=I+4THEN260
250 PRINT'I';:M=M+1:GØTØ280
260 NEXTJ
270 !DØES THE ANSWER HAVE FØUR CHARACTERS?
280 NEXTI:IFM=4THEN160
290 !RØUND ØUT THE ANSWER
300 M=M+1:FØRI=MTØ4:PRINT'-';:NEXTI:GØTØ160
310 !START ØVER AGAIN
320 PRINT' YØU DID IT IN';N;'MØVES
      LETS DØ IT AGAIN.';:GØTØ130
```

Sample Run

```
? 1234
II--
? 1256
XI--
? 5555
XIII
? 6666
----
? 2222
XXII
? 2152
XXX-
? 2352
XXX-
? 2452
XXXX YØU DID IT
   IN 8 MØVES LETS
   DØ IT AGAIN.

? 1234
X---
? 2222
XXII
? 1221
XI--
? 2211
XX--
? 2251
XXX-
? 2256
XXX-
? 2255
XXXX YØU DID IT
   IN 7 MØVES LETS
   DØ IT AGAIN.
```

The Computer Solves the Four Color Map Problem



BY WILLARD E. MATHESON

Early in the 1960's a second generation of computers emerged in which the transistor replaced the vacuum tube of previous designs. The increased speed and reliability of the new computers provided tools essential to the newly created national space program. Indeed the synergy which developed in that decade between computers and systems analysis in the space and military sciences was responsible for the phonomenal improvements in hardware and software and the rapid expansion into scientific and business fields.

Scientific applications of the computer burgeoned in the 60s, primarily in the area of the applied sciences and physics. A new society of engineers and physicists trained in use of the computer appeared on the scene. Unfortunately, mathematicians, – especially theoretical mathematicians, – remained generally apart from the computer world at first.

In the 1970s though, the computer finally began to stir the interests of theoretical mathematicians. The pure mathematician's admitted lack of training in computers inhibited his effective participation in the field, and this unfamiliarity accounted for the mathematician's mistrust of results which could not be verified independently of the computer.

The most outstanding example of successful use of a computer to a previously unsolved problem in mathematics is the famous "four-color map problem". The computer-aided proof of

the century-old conjecture was completed in 1976 and published in 1977, by Professors V. Appel and W. Haken at the University of Illinois. This proof cannot in fact be checked by hand calculations. In over a century of search no simple elegant proof of this problem has ever been demonstrated. But the search for such a proof has stimulated development of new branches of mathematical science in the fields of combinatorial mathematics and topology.

The four-color map problem dates back to 1852 when an Englishman, Francis Guthrie, observed to his brother, a student of famed mathematician Augustus De Morgan, that every map drawn on paper and colored in a manner such that adjoining countries have different colors seemed to require only four colors. Guthrie conjectured that this might be proven mathematically.

The search for proof that four colors were sufficient challenged and frustrated mathematicians for an entire century since Guthrie's observation.

In England the problem appeared to be solved in the 1879 publication of a paper by barrister-mathematician, Alfred Bray Kempe. But an error in the proof was demonstrated by another English mathematician Percey John Heawood in 1890. Although Heawood spent 60 years on the problem, 86 years more elapsed before the four-color solution finally yielded to the work of mathematicians Appel and Haken at the University of Illinois in 1977. The two had worked on the problem from 1972 to 1976 searching for an extension of the correct portions of Kempe's proof.

Two amazing revelations appeared in the 1977 papers of Appel and Haken. The first was that these two mathematicians had finally solved the problem that had frustrated all attempts for one and a quarter centuries. The second was even more significant in its implications to the world of mathematics: the successful proof was dependent upon the use of a high-speed computer and could not have been accomplished without it. Furthermore, the proof could not be checked without the use of the computer. This was indeed disturbing to mathematicians. Generally untrained in the use of computers, pure mathematicians as already stated, do not trust results unless they have an independent method of checking. The computer-assisted proof of the four-color conjecture demanded about 10 billion logical decisions and required more than 1200 hours analyzing thousands of configurations by computer. These calculations could not have been accomplished without the computer in less than 100 human lifetimes.

Prior to the successful completion of a computer proof some success was achieved by more conventional methods. In 1913 Harvard mathematician Birkhoff extending the results of Kempe; and Franklin, an M.I.T. mathematician, used these results to show that any map with fewer than 22 countries is four-colorable. By 1976 this result had been extended by other mathematicians to 92 countries.

To learn what was accomplished by the computer we must look at a more exact statement of the four-color map problem. Adjoining countries are defined as those sharing a border of finite extent. Multiple touching at a point is excluded, otherwise as many colors would be required as there are countries having the point in common. Furthermore, a country is interpreted to mean a single connected region. (Otherwise a five-color map is easily constructed.) These clarifications are necessary to make the conjecture a logical one. Examples of these restrictions are afforded in the U.S. map of the 48 conterminous states – which is not a proper map. A violation is seen in the four-corners area where four states meet at one point. Other violations of the requirements for a proper map occur in the separate mainland portions of the state of Michigan, the state of Washington, and in all states with island possessions.

Early researchers of the four-color map problem recognized that a three-color conjecture is easily shown to be false (see Figure 1). Also De Morgan proved that it was impossible for five countries each consisting of a single connect region to be each adjacent to the other four. The fact that this proof is not equivalent to proving the conjecture may be seen by comprehension of the mathematician's formulation of conditions and examination of Figure 1.

In analyzing the requirements for a proposition to be logically true, mathematicians look for what they call necessary and sufficient conditions. Clearly, conditions may be found which are essential to the truth of the proposition, and the identification of

The successful proof was dependent upon the use of a high speed computer and could not have been accomplished without it. Furthermore, the proof could not be checked without the use of the computer.

some of these conditions may be far from a trivial exercise. The existence of some of these conditions may not, however, be adequate to validate the theorem under all circumstances. Precisely the right number of conditions must occur to satisfy the "sufficiency" condition of a proof. On the other hand, superfluous conditions which are not under all circumstances required for proof are not "necessary", and their stipulation offends the mathematicians' search for minimum requirements and maximum simplicity.

The search for the solution to the four-color map problem is equivalent to looking for a proof that four colors are necessary and sufficient for the map as defined. In Figure 1 it is easily seen that four colors are necessary; the difficulty is to show that four are sufficient. Heawood had shown the sufficiency of five colors in 1890, Kempe formalized the problem by defining a normal map as one in which no country encloses another and in which no more than three countries meet at a point. He looked for a proof of the theorem using normal maps since he was able to show that a proof valid for normal maps would be valid for all. He proved that the very existence of a "five-chromatic" map (requiring five colors) would require the extence of normal five-chromatic map, and that the existence of the latter would imply the existence of such a map with the smallest possible number of countries. This he called a minimal normal five-chromatic map (MNFC map).

With these definitions it is seen that any map with fewer countries than the number characteristic of the MNFC map would require at most four colors. Kempe had thus defined the search for a proof that "a minimal normal five-chromatic map is impossible." In other words any attempt to prove the existence of a minimal normal map demanding five colors would inevitably lead to a contradiction of the existence postulate.

Kempe's first steps to a proof were sound and remained as the underpinnings of the proof completed by Appel and Haken almost a hundred years later. Step one was his proof that any normal map must contain at least one country with less than six neighbors. Since this is true for any normal map it must be true also for a MNFC map. Kempe's proof assumed the existence of MNFC map and looked for a logical contradiction of the assumption which would prove the theorem: namely, that the existence of a minimum normal map with fewer countries that is also five-chromatic. To do this he had to show that a contradiction is arrived at whether the number of countries in the MNFC map is two, three, four, or five neighbors. (The zero or one neighbor cases are excluded by the definition of a normal map, since neither an island nor an enclosed country is permitted.)

Kempe proved the case for the two, three and four neighbor configurations but failed in the case of five neighbors (see Figure 2). The flaw in his proof of the five neighbor case, discovered by Heawood eleven years after Kempe published his proof, was eventually repaired by Appel and Haken with the aid of a computer unavailable until the latter part of the 20th century.

The four configurations found by Kempe to be unavoidable in a normal map came to be known as an "unavoidable set of reducible configurations.

Another quarter of a century was to elapse before the methods of finding unavoidable sets and of determining reducibility were fully developed. Again, Heesch provided the important keys in his development of essential methods of describing configurations and determining reducibility. Crucial to the final solution was his introduction of an electrical analog method called "discharging" for finding unavoidable sets of configurations.

Heesch with his student Durre was the first to apply the computer to the problem of proving configurations reducible. His method involved transforming the map into a "dual graph" of edges and vertices. These ideas and procedures were sufficient to provide the reducibility methods finally used by the computer.

However, although Heesch developed the method of discharging for finding unavoidable sets of configurations, these procedures were not sufficiently far enough along in the late 60s for the solution of the problem to be clear.

In 1970 Haken found methods of improving the discharging procedures

Following is a recent exchange of correspondence between Willard Matheson, author of this article, and Kenneth Appel, one of the solvers of this century-old problem:

Dear Dr. Appel:

One thing I would like to ask about concerning extensions of Birhoff's methods. You mentioned that by 1950 any map with fewer than 36 countries had been shown to be four-colorable. Is this statement up to date? Although I have not consulted the mathematical journals I am speculating that this had been extended prior to composition of your article. Hogben in "Mathematics in the Making" (1966) states that the sufficiency of four had been shown for not more than 38 regions, and the 1977 Britannica indicated that four colors had been demonstrated to be sufficient for fewer than 40 countries.

Sincerely,
W.E. Matheson

Dear Dr. Matheson:

The reason we mentioned 36 was because of the early date of Winn's work (prior to 1940). The next significant advances were: Ore & Stemple (1970) 40; Stromquist (1975) 52; J. Mayer (1975) 72; and again J. Mayer in 1976, 92. I think the 38 you mention (Hogben estimate) was not a correct piece of work, but I am not certain.

Sincerely,
Kenneth Appel

and made calculations of the feasibility of finding an unavoidable set of reducible configurations small enough to be handled in a reasonable number of years using the largest computers. When Haken and Appel began their search in 1972 they were convinced that a non-computer proof was not forthcoming with known techniques.

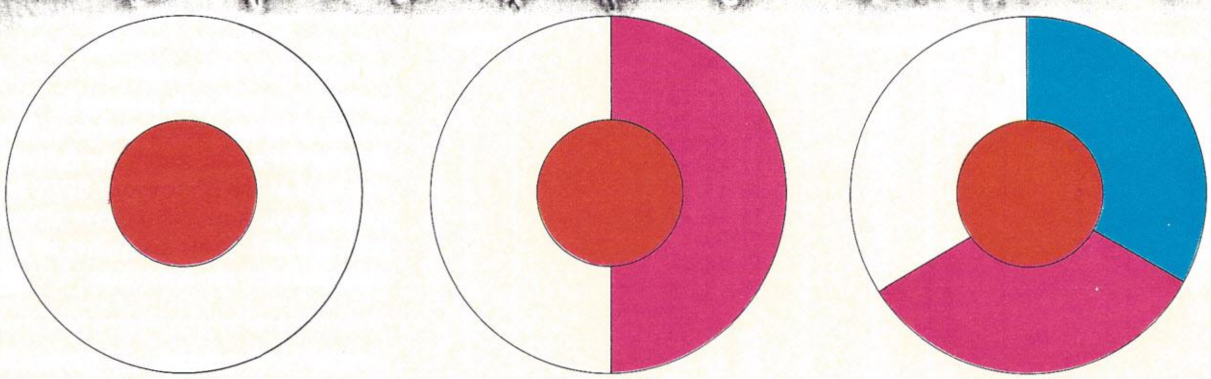
Concentrating on computer techniques to discover a finite unavoidable set of reducible configurations, the authors steadily improved their techniques and computer programs for several years. By 1975 the authors had their computer operating in a learning mode like a chess-playing machine. They stated that the computer "was working out compound strategies based on all the tricks it had been taught, and the new approaches were much cleverer than those we would have tried". They concluded that "the program was demonstrating superiority not only in the mechanical parts of the task but in some intellectual areas as well."

Another year's work on improving discharging methods and the refining of programs for reducibility brought the effort to the brink of solution. Finally, in June 1976 the work was finished with the completion of construction of an unavoidable set of reducible configurations. The reduction of 1482 configurations was included in the final proof. It is apparent that complete checking of the proof is impossible without using another computer.

What are the implications of this proof of a century-old problem? The long proof is disturbing to many and departs from mathematical tradition. A short proof of the problem may someday be found but it is certainly not known whether this will happen.

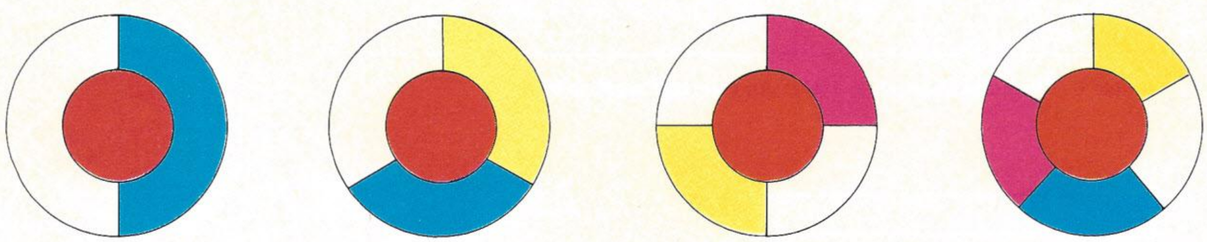
Appel and Haken believe that there exist theorems of important mathematical interest which will be solved only through the use of the computer, and that there is every reason to expect that the number of problems requiring such methods is large. They state that the proof of the four-color problem suggests that there are limits to the power of theoretical methods alone in mathematics and believe that mathematicians should continue to explore the powers and limitations of their methods.

Stanislaus Ulam, the mathematician who developed at Los Almos the mathematics of the hydrogen bomb, predicts that future mathematicians will occupy themselves with large-scale problems rather than with such details as special theorems, and that computers will be essential in guiding and assisting their work.



One neighbor mutually adjacent | Two neighbors mutually adjacent | Three neighbors mutually adjacent

Figure 1 The one- neighbor case is excluded by the definition of a normal map which does not permit a country to be totally enclosed by another. It can be seen from the diagram that although three colors are sufficient for the two-neighbor case four are necessary for the three-neighbor case.



2 neighbors | 3 neighbors | 4 neighbors | 5 neighbors

Figure 2 The configurations of two, three, four and five neighbors mutually adjacent in a normal map comprised Kempe's set of unavoidable configurations. He showed the first three to be reducible but failed to find a valid proof for the five-neighbor case.

How the Computer Attacks the Four-Color Problem

Maps are colored to make it easy to see, at a glance, the extent of each country. It is clearly necessary that neighboring countries be assigned different colors. Does the map maker then need more than four colors to do his job? He doesn't care, but we do.

The problem is one of the most celebrated challenges in mathematics. It is of great intellectual interest and has intrigued many people from all paths of life. Its solution has little or nothing whatsoever to do with making maps. A map maker is and always will be able to print maps using as many different colors as he needs.

A checkerboard is an example of a map that can be colored with only two colors. A four-country map requires four colors. The reason for this is that, each pair of countries being adjacent, no two can have the same color.

The problem of four-coloring a given map is one of traveling along a path through a tree like that shown in Figure 1. Each segment represents a decision to color a country with colors 1, 2, 3 or 4. The ith segment in a path from the root corresponds to the coloring of the ith country of the map

We can always chose three countries, which are neighbors of one another and begin by coloring these countries. Since each of these countries must then be of a different color and it is immaterial which colors we assign them, we choose colors 1, 2 and 3. That is why our tree shows only one segment coming from each of the first three nodes. In coloring all the remaining countries, it's convenient to imagine that for each there are up to four choices possible. Most of the time, however, only one, two or three choices will be admissable. Sometimes all four choices will be inadmissable.

How do we represent a map in computer memory? One way is to construct a "connection table" listing after each country all of its neighbors in increasing order. Our algorithm can consult this information when deciding how to color a particular country. For example, if we were coloring country number 15 in Figure 2, we could see in row 15 that countries numbers 5, 6 and 14 are neighbors that have been colored already. Our choice of colors for number 15, then, depends solely on the currently chosen colors for countries 5, 6 and 14.

Knowing that country number 15 also has neighbors numbered 16, 25 and 26 appears to be superfluous to our needs. This leads to the idea of a "shaved-down" table which we call the *"reduced* connection table." It is constructed by striking out of each row in the table all numbers greater than the number of the row itself. The "reduced connection table" is seen in Table 1 and can be thought of in this case as a 39 row of 7 columns array called *CONN*. The number of nonnull elements in each row is given by elements of an associated vector w. Thus, the algorithm can search the first wi elements in the ith row of *CONN* to determine which neighbors have already been colored. Table 1 supplies all the input data for the algorithm. ▮◖

Adapted and reprinted, by permission, from Forsythe's, et al, "Computer Science, First Course" second edition. Pages 334 and 335. Copyright ©1975 by John Wiley & Sons, Inc. New York, NY.

Country i	Neighbors $CONN_{ij}$	Width w_i	Country i	Neighbors $CONN_{ij}$	Width w_i
1		0	21	10 11 20	3
2	1	1	22	11 12 21	3
3	1 2	2	23	12 13 22	3
4	1 3	2	24	13 14 23	3
5	1 4	2	25	14 15 24	3
6	1 2 5	3	26	15 16 17 25	4
7	2 6	2	27	17 26	2
8	2 7	2	28	17 18 27	3
9	2 3 8	3	29	18 19 28	3
10	3 9	2	30	19 20 29	3
11	3 5 10	3	31	20 21 30	3
12	4 11	2	32	21 22 31	3
13	4 5 12	3	33	22 23 32	3
14	5 13	2	34	23 24 33	3
15	5 6 14	3	34	24 25 34	3
16	6 7 15	3	36	25 26 27 35	4
17	7 16	2	37	27 28 29 36	4
18	7 8 17	3	38	29 30 31 32 37	5
19	8 9 18	3	39	32 33 34 35 36 37 38	7
20	9 10 19	3			

Table 1 "Reduced Connection Table" for coloring "map" in Fig. 2.

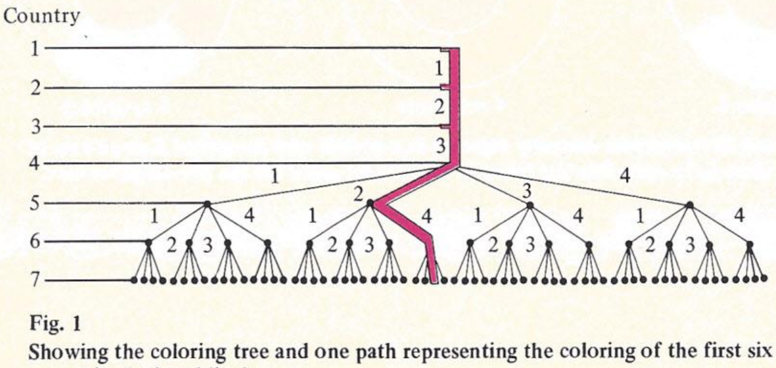


Fig. 1
Showing the coloring tree and one path representing the coloring of the first six countries (colored line).

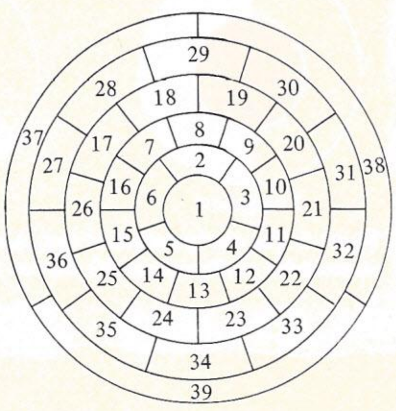


Fig. 2 Example of map to be four-colored by a computer algorithm.

Waltzing Your Micro to the PROMs

BY CHIP A. TYETI

Spring is here, and as we PROMenade down Memory Lane, we're overwhelmed by the profuse proliferation of PROMs in pertinent peripherals pervading the pathway (and all that pizzazz). Companies offer (E)PROM programmers, PROM memory boards, CPU boards with on-board PROMs and video display boards with PROM, in addition to floppy disk controllers with PROM, specialty boards with on-board PROM, and UV lamps for erasing PROMs.

Let's start with basics. ROM simply means "Read Only Memory". You can execute or dump from anywhere in the ROM chips' memory locations; you cannot write into memory, though. ROM chips included in kits and assembled components are primarily ROM chips pre-programmed by the semiconductor house, who then sell them to the manufactuer from whom you buy the units. These factory-programmed chips cannot be reprogrammed. Masking and photolithographing the circuit into the ROM chip makes it an integral part of the chip.

PROM chips differ; they're usually programmed by the company that sells you the system. In other words, the semiconductor house ships blank ROMs, to the system manufacturer who programs them with custom software packages.

Although PROM chips in the strict sense cannot be erased, specific chips vary in reprogramming capability. That is, academically you should call a custom-programmed chip a PROM if it comes from the systems manufacturer, but technically some chips are also eraseable and are therefore EPROMs.

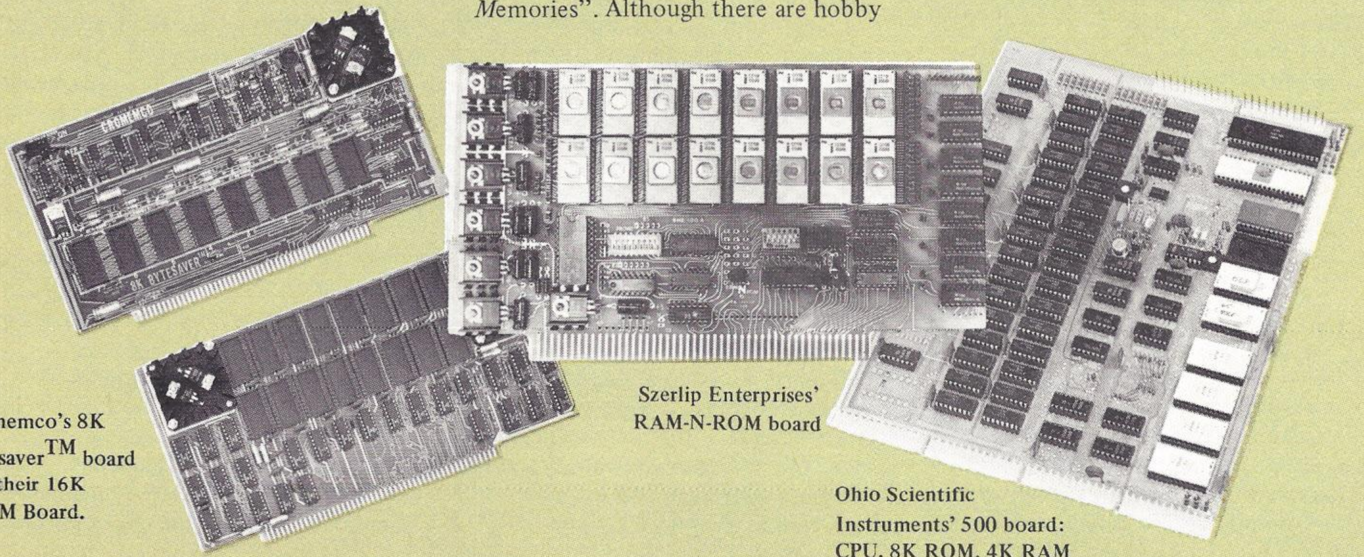
You would probably buy EPROMs to plug into a homebrew, commercial or hobby PROM programmer. An EPROM chip may be erased several times (perhaps as few as a dozen or as many as a hundred) and reprogrammed. One very handy device erases EPROMs in about 10-15 minutes. This compact, efficient ultraviolet lamp is valuable for anyone actively involved in software development.

EAROM is the final variation of ROM we'll mention here. EAROMs are "*E*lectronically *A*ltered *R*ead *O*nly *M*emories". Although there are hobby kits available that can program your EAROMs, we'll not go into them here.

Since so many boards and units involve ROMs, PROMs or EPROMs, we'll discuss specifics of a few. The chart lists almost fifty different boards/units, some of which are shown in the photographs below. The column "Chip Used" refers to the chips that can be programmed, if the device is a PROM programmer. If the device is a PC board, this column lists the chip used in the specific circuit. The PROM feature may be the most important (or perhaps only) feature of the device; if not, secondary uses are mentioned. Note that not all devices come in both kit and assembled versions. Many manufacturers are dropping kits from their listings, offering assembled and testing units backed by reasonable warranties.

You can buy nearly all units off the shelf, but a few units need two to three month lead time. We've listed RAM capacity of units available with RAM.

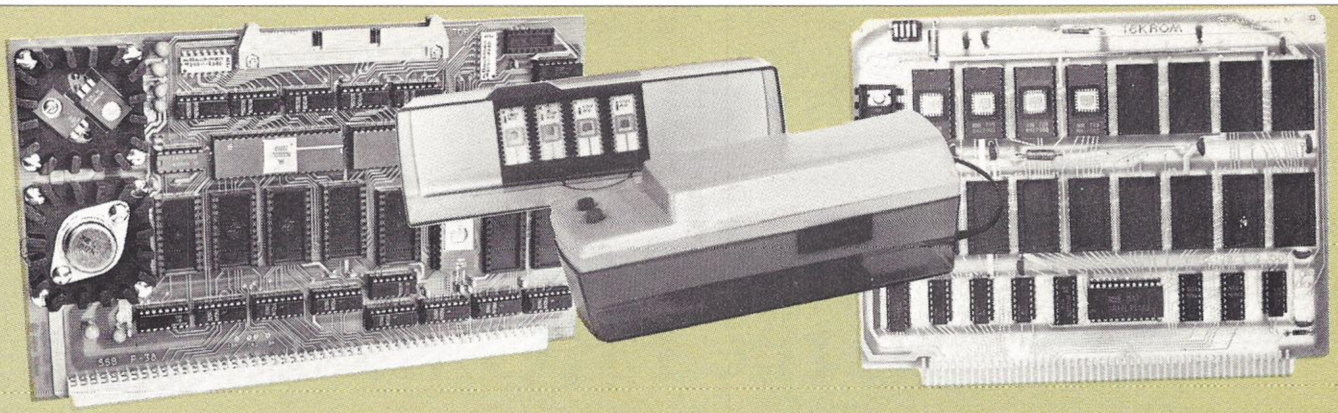
Advanced Microcomputer Products, Miniterm Associates, North Star Computers, Szerlip Enterprises, Vector



Cromemco's 8K Bytesaver™ board and their 16K PROM Board.

Szerlip Enterprises' RAM-N-ROM board

Ohio Scientific Instruments' 500 board: CPU, 8K ROM, 4K RAM



Smoke Signal Broadcasting's P-38-I   Ultra-Violet Products' UVS-11E EPROM Erasing Lamp   GNAT Computers' 16K PROM board

Graphics, Solid State Music and several other manufacturers configure their PROM boards with a power-on/reset/jump command. Once you turn on your microcomputer, the system jumps to the PROM location for immediate execution.

Katherine Atwood Enterprises offers an EPROM Programming Board that can copy a PROM. If you want to make back-up copies of your firmware or market that unbelieveable software package, a PROM copier might come in handy. The Gimix 4KPPD Board can write all four PROMs at the same time, making mass production somewhat more efficient.

Most units listed are compatible with 2708s and/or 2716s, but many include a larger selection of compatible chips. Szerlip Enterprises' RAM-N-ROM board and the PROM Setter device will accept the largest array of chips.

Check the PROMs with which you might be experimenting; make a list of PROM chips to check against the equipment you are evaluating from this listing.

Apple Computer's main PC board contains 12K of PROM in addition to 48K of RAM. The 9316 chip used on the board has the Apple 50 pin bus structure, of course.

Cromemco sells three boards that use PROMs rather extensively. The 32K Bytesaver memory board offers switches that provide for protect and unprotect of PROMs individually or in groups, shadow ROM socket pairs and select card address. In addition, this board offers bank select switching and DMA IN–OUT features. The 32K Bytesaver not only holds 32K PROM, but also programs 2716 PROMs. This board looks like a versatile, economic value for S100 bus computerists.

Cromemco's second board, the 16K PROM card, also includes the bank select feature (allowing the computer access to up to 8 banks of 64K memory each). The board also features address anticipation, so there are no wait states required in the usual sequential addressing operation.

The third board offered by Cromemco is the Bytesaver memory board, which is both a place and a way to store programs. The built-in programmer is designed for 2708 PROMs. With 2708s in all 8 sockets, the Bytesaver still draws only about 500 mA from the +8V bus.

Dajen Electronics' board, the SCI, is a serial and parallel I/O board and a cassette I/O board, and has 3K worth of PROM on-board; 2K of the PROM includes an advanced monitor program. This board is worth investigating because it has an extensive list of features beyond those mentioned here.

Digital Research Corporation of Garland, Texas, offers their 16K EPROM card at a reduced price of $57.50 assembled – about a 40% less than their original price. It's a bargain buy in 16K PROM boards!

Dynabyte's Z80 Single Board Computer includes either 2 or 4 MHz operation, two RS-232 serial I/O ports, one parallel I/O port, 4K of EPROM with programming, 1K of static RAM, vectored interrupts, a real-time clock, and power on/jump – all in an assembled, tested and burned-in S100 PC board warranted for one year. Dynabyte's Z80 SBC features deselection capabilities of the EPROMs on board, so they won't be occupying address space.

Gimix offers three boards with PROM capacity: 8K EPROM, the CPU, and the 4K PPD. The 8K EPROM uses 2708 UV EPROMs and is DIP switch addressable to any 8K memory boundary. It can be addressed to E000 to replace MIKBUG using your own system monitor. Instructions included allow minor CPU board modifications for this purpose.

Gimix's CPU board includes Xtal-controlled 6800 CPU, a baud rate generator for 110-9600 baud (separate crystal), 6840 programmable timer, 6810 RAM of 128 bytes, and four sockets for 2708 PROMs. This CPU has DIP switch selectable addressing (including E000/FC00 dual access for MIKBUG. The board has fully buffered address and data buses and DMA capability through cycle-stealing or Halt.

The third Gimix board, the 4K PPD, is a PROM board with a built-in programmer and duplicator. It holds 2708s (one is included) and features DIP switch addressing to any 4K boundary. A master switch removes power from the PROMs, allowing insertion and removal of ROMs without shutting down the computer. There are separate write protect switches, one for each PROM. All four PROMs are written at the same time, but are read separately. The one PROM included with the 4K PPD contains writing and testing software. 6800 users should check out these boards.

D.C. Hayes Associates lists their 80-708 Reliable 8K EPROM memory board containing power regulators that operate well below their rated current even in the worst case. High reliability ROM sockets on board can withstand many more insertions than typical sockets. All address and control lines have buffers with input hysterisis. Two separate 4K bytes of ROM/EPROM can be independently located on any 4K boundary by setting the hex digit of the address. Setting both address switches to the same value disables the four ROM sockets on the right side of the board. The entire board occupies only 4K of memory space.

Micro Systems Development markets the MSDV-100 Video Display System board for S100 bus computers. This board offers a wide variety of features for graphic display of characters, charts and graphs. The character gen-

Szerlip Enterprises' PROM Setter **Smoke Signal Broadcasting's POP-1 Programmer** **RDA's RMRV-8K EPROM board**

erator in the board is actually a pair of high-capacity ROM devices compatible with the C2708 chip. Thus, any user with access to a programmer for these devices can easily invent a custom character set. The set contains 128 separate positions, which can be defined as any character that will fit on a 6x10 matrix. Storing PROM memory is a secondary function of this board, of course.

Miniterm Associates' ROM/EROM board contains the bank select feature discussed earlier. Miniterm's board allows memory expansion to 128K, and functions as two separately addressable memory boards. Each memory position may be individually enabled or disabled to the CPU bus. This feature allows other memory devices to be addressed within the same memory block and allows the ROM/EROM board to provide as little as 1K of ROM at any 1K address boundary. You could let this board and your floppy disk controller share the same memory block. Miniterm provides superb documentation – a comprehensive 22-page manual which details construction, use and theory of operation. Ideal for turnkey systems and front-panel-less systems, this board can serve as a substitute for systems with front panels to allow jump start capability. Check it out.

Mountain Hardware manufactures the PROROM board, an 8K EPROM memory board including ½K RAM for a stack or scratch pad on-board. Note that this board uses AMI S-6834 EPROMs which allow programming any number of bytes at a time. Erasing fills the 6834 with 00s – NO OP instructions to an 8080. A fully stuffed PROROM costs 2 cents per byte; 2708s cost 2.5 cents per byte. Thus, the PROROM board is 25% less expensive than boards using 2708s. Programs as large as 7½K can be loaded from PROROM in less than one second. Any of the EPROMs can be erased in about 15 minutes using an inexpensive ultraviolet light.

Ultraviolet Products, San Gabriel, CA, offers their UVS-11E low-cost EPROM erasing lamp for economical, efficient and safe EPROM erasing. A special interlock system, complying with the National Consumer Safety Act, prevents use unless the cover is secured in the closed position. UV light can damage retinas, and this feature makes this unit handy, yet safe and affordable for small systems users and hobbyists.

The UVS-11E holds up to four EPROMs in special conductive foam, weighs about a pound and costs a mere \$59.50, with replacement parts always available from stock. Ultra-Violet Products has a complete line of UV lamps and safety equipment for professionals.

To date, the question of accumulative UV damage to eyes has not been answered. Nevertheless, be very careful when exposing your eyes to UV light.

Ohio Scientific Instruments' Model 500 board can serve as a CPU, or store 8K BASIC in ROM and be used as a 4K RAM board. Because it can hold instant BASIC and RAM, it can serve as a powerful small system or as the basis of a larger system. It supports additional memory and other peripheral devices offered by Ohio Scientific.

A second board from Ohio Scientific, the 450 B 8K EPROM and Parallel I/O board, can provide permanent storage and parallel I/O. It features an on-board programmer for 6834s, among the most cost effective EPROMs available.

A third board with PROM by OSI is the 455 4K EPROM and parallel I/O board, which provides permanent storage and 16 lines of parallel I/O. Check the photos for comparison. All three use the OSI 48-pin bus.

Oliver Advanced Engineering sells two PROM programmers in elegant small packages. One (the PP-2708/16) programs 2708s and TMS 2716s; the other (the PP-2716) programs Intel 2716s. Plugging either one into a PROM socket gives you instantly all the features of even the most expensive programmers. Both units use zero insertion force sockets and a short software routine which sends data over the eight lower address lines using OAE's unique interfacing technique. No additional power supplies are required and all timing and control sequences are handled by the programmer. Multiple programmers connected in parallel allow gang programming. The programmers can also interface to an 8-bit parallel port.

Optimal Technology has an EPROM programmer for several micros: the Intel 8080, Motorola 6800, MOS Technology 6502, Fairchild F-8, and RCA 1802. The programmer offers on-board provisions for programming both 2708 and 2716 EPROM. The microcomputer monitors and specifies any RAM starting address up to 65K. Also, any PROM starting address within the address space of the PROM may be specified along with the number of bytes to be programmed. A verify mode confirms that all bits have been programmed correctly. Price variations on the chart reflect the type of programming socket.

Processor Technology uses PROM in their GPM modules which correspond to the SOL single-board computer. GPMs give owners of non-SOL microcomputers a chance to experience the ease of operation of the SOL and the SOLOS firmware. Processor Tech indicated a while back that someday they will offer a wide variety of personality modules for the SOL, each using a variety of one to four PROM chips. When a selection of personality modules (other than CONSOL, SOLED, and SOLOS) becomes reality, effective

PROMSpecs – A Comparative Chart of

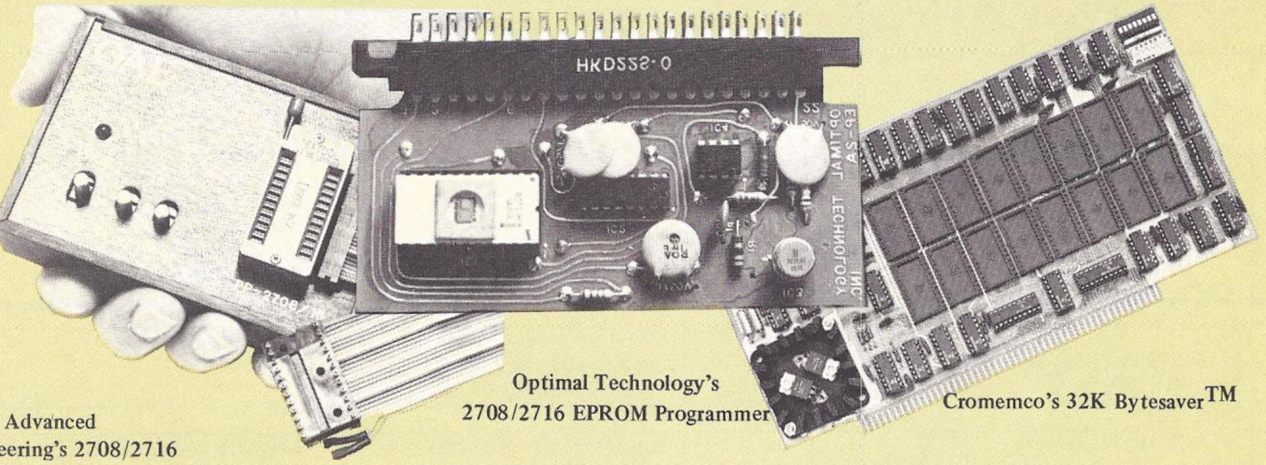
Company Name	Mail Order/Retail/	Warranty Period	Board Name/Number	Kit Price	Assembled Price	Bus Structure	Availability	PROM/ROM Capacity (K)
Advanced Computer Products	MO	120d	Byteuser	65.	100.	S100	1	8K
Katherine Atwood Enterprises	MO	[4]	EPROM Programming Board	50.	NA	S44	1	0
Katherine Atwood Enterprises	MO	[4]	8K EPROM Board	150.	NA	S44	1	8K
Base 2	MO	90d	16Kx8bit ROM	[1]	[1]	S100	2	16K
Compucolor Corporation	RE	[4]	24K ROM/PROM Card, Option 20	NA	100.	[17]	1	24K
Cromemco	RE	90d	32K Bytesaver(TM)	195.	295.	S100	3	32K
Cromemco	RE	90d	16K PROM Card	145.	245.	S100	1	16K
Cromemco	RE	90d	Bytesaver(TM) Memory Board and PROM Program'r	145.	245.	S100	1	8K
Dajen Electronics	MO	[4]	SCI	285.	345.	S100	1	3K
Digital Group	RE	90d	1702 EPROM Memory Board	299.	345.	DG	1	4K
Digital Research Corporation	MO	[4]	16K EPROM Board	NA	58.	S100	1	16K
Dynabyte	RE	1y	Z80 Single Board Computer	NA	555.	S100	2	4K
Electrolabs	MO	[4]	MEM-11-24+2	NA	599.	LSI11	1	2K
Electronic Product Associates	RE	90d	PROM 16K	NA	264.	Exorc'z'r	1	16K
Electronic Product Associates	RE	90d	48K Memory Board	NA	385.	Exorc'z'r	1	32K
Gimix	RE	[4]	8K EPROM	NA	[2]	SS50	1	8K
Gimix	RE	[4]	CPU	NA	[2]	SS50	1	4K
Gimix	RE	[4]	4K PPD	NA	[2]	SS50	1	4K
Gnat	RE	[4]	8020, 4K ROM	NA	145.	GNAT	1	4K
Gnat	RE	[4]	8025, 4K ROM/RAM	NA	145.	GNAT	1	2K
Gnat	RE	[4]	8021, 16K ROM	NA	145.	GNAT	1	16K
D. C. Hayes	MO	[4]	80-708, Reliable EPROM Board	NA	163.[3]	S100	1	8K
Ibex	RE	[4]	16K PROM Board	85.	110.	S100	1	16K
IMSAI	RE	1y	PROM 4-4	399.	579.	S100	1	4K
IMSAI	RE	1y	PROM 16	[1]	[1]	S100	2	[4]
Micro Systems Development	MO	[4]	MSDV-100	285.	385.	S100	2	4K
Midwest Scientific Instruments	RE	90d	PRR-68	95.	150.	SS50	1	<4K
Miniterm Associates	RE	[4]	ROM/EROM Board	89.	129.	S100	1	16K
Mountain Hardware	RE	[4]	PROROM Board	164.[5]	214.	S100	1	7½K
North Star Computers	RE	90d	ZPB-AW/EPROM Option 6	248.	328.	S100	2	1K
Ohio Scientific Instruments	RE	60d	500	[7-8]	298.	OSI48	[4]	8K
Ohio Scientific Instruments	RE	60d	450B 8K PROM and Parallel I/O Board	[7]	[4]	OSI48	[4]	8K
Ohio Scientific Instruments	RE	60d	455 4K EPROM and Parallel I/O Board	[7]	[4]	OSI48	[4]	4K
Oliver Advanced Engineering	RE	[4]	PP-2716 and PP-2708/16 PROM Programmer	249.	295.	[9]	1	NA
Optimal Technology	RE	[4]	EP-2A-K, -M, -R, -F, -8I	[18]	[19]	[10]	1	½-2K
PerSci	RE	[4]	1070 Floppy Disk Controller	NA	740.	72Vctr	1	4K/E
Problem Solver Systems	RE	120d	MP12 Programmer	NA	145.	S100	2	8K
Processor Technology	RE	90d	GPM	[20]	[21]	S100	1	10K
RDA	RE	[4]	RMRV-8K	NA	285.	Qbus	1	16K
S.D. Sales	MO	90d	EconoPROM	50.	[11]	S100	1	32K
Seals Electronics	RE	90d	4K ROM	109.	119.	S100	1	4K
Smoke Signal Broadcasting	RE	90d	POP-1	NA	129.	SS50	1	0
Smoke Signal Broadcasting	RE	90d	P-38-I, -FF	129.	174.	SS50	1	8K
Solid State Music	RE	90d	4K Static PROM/RAM Board	80.	105.	S100	1	4K
Solid State Music	RE	90d	MB-8 8K/16K EPROM Board	75.	100.	S100	1	16K
Solid State Music	RE	90d	MB-3 2K/4K EPROM Board	65.	90.	S100	1	4K
Southwest Technical Products	RE	90d	MP-A2 PROM Board	145.	175.	SS50	1	8K
Space Byte	RE	90d	8085 CPU	NA	499.	S100	1	6K
Synertek Systems	RE	[4]	MM-200	NA	149.	Jolt	2	2K
Szerlip Enterprises	RE	[4]	RAM-N-ROM: RNR-100A	117.	169.	S100	2	32K
Szerlip Enterprises	RE	[4]	The PROM Setter	210.	375.	S100	2	0
Technico	RE	[4]	Super Starter System T9900-SS	299.	399.	[16]	1	4K
Technico	RE	[4]	E-PROM Board T9900-EP	100.	150.	[16]	1	28K
Technical Design Labs	RE	90d	System Monitor Board II	NA	395.	S100	1	2K
Vector Graphics	RE	90d	PROM/RAM Board 1702	89.	129.	S100	1	2K
Vector Graphics	RE	90d	PROM/RAM Board 2708	135.	175.	S100	1	12K

FOOTNOTES 1 consult mfr. for price; 2 consult dlr. for price; 3 less EPROMs; 4 consult mfr. for specs; 5 incl. one preprogrammed EPROM; add'l unprogrammed EPROMs @ $10 ; 6 incl. one PROM; 7 $39 : bare board and manual; 8 $134 (504V) incl. 65V PROM monitor; $149 (504A) incl. 65A PROM monitor; 9 unique interfacing technique allows easy interface w/ most μp based systems

PROM Boards and PROM Programmers

RAM Capacity	Chip(s) Used	Can it program a PROM?	Is the PROM feature the primary one?	Is there an additional use for the unit?	ADDITIONAL COMMENTS
0	2708	no	yes	no	power on jump/reset jump
0	2716, 2708 with modification	yes	yes	no	has a PROM copy mode
0	2716 (5 V only!)	no	yes	no	8K PROM can reside in 4K of memory
0	2708, 2716	no	yes	no	-
0	2708, 2716, 9316s	no	yes	no	will accept any combination of ROM chips with only minor jumper changes
0	2716	yes	no	yes	is also a 2716 programmer – 4 switches for on-board control
0	2708	no	yes	no	bank select; address anticipation feature
0	2708	yes	no	yes	memory board draws only 500 mA from +8V bus
256B	2708	yes	no	yes	cassette I/O; serial or p'll I/O; for 8080 or Z80
0	1702	no	yes	no	can be purchased blank for $35.
0	2708	no	yes	no	-
1K	2716	yes	no	yes	single board computer; I/O, PROM, RAM, RTC, programming capability
24K	2708	no	no	yes	the only LSI-11 compatible memory board with PROM
0	generic 4K PROM (e.g. Intel 3614)	no	yes	no	a fusible link PROM board
16K	2708, 2716; 2114 RAM	no	yes	no	largest capacity single static RAM and EPROM board
0	2708	no	yes	no	DIP switch addressable
0	2708	no	no	yes	CPU
0	2708	yes	yes	yes	a programmer also: all 4 PROMs can be written @ same time for dupes
0	1702	no	yes	no	-
2K	1702	no	yes	no	-
0	2708	no	yes	no	selectable for 8K or 16K/2708s; also can use 2704s;mostly comm'l use
0	2708	no	yes	no	(cf. text of this article)
0	2708	no	yes	no	can be enabled and disabled in 4K increments
0	2708	no	yes	no	-
0	2708	no	yes	no	-
2K	2708	no	no	yes	video controller (cf. text)
256B	1702	no	yes	no	also uses 8K EPROM:2708
0	2708, 2716	no	yes	yes	restart/jump – front panel replacement
½K	6834	yes	yes	no	(cf. text)
0	2708	no	no	yes	Z80 CPU, incl. interrupt priority decoder; 16 bit power on-jump circuit
4K	[4]	no	no	yes	CPU, 4K RAM board (cf. text)
0	[4]	yes	yes	no	(cf. text)
0	[4]	no	yes	no	16 lines of parallel I/O
NA	all 2708s, TMS 2716, Intel 2716	yes	NA	no	plugs directly into any PROM socket; driving s/w: short+simple;(cf.text)
0	2708, Intel 2716	yes	no	no	complete s/w for F-8, 6800, KIM-1, 8080, 1802
1K	2708	no	no	yes	has 8080 on-board; fully intelligent - talk to it in ASCII commands
0	2708	yes	yes	no	(cf. text)
1K	2708, 9216, 8316, 34000	no	yes	no	(cf. text)
0	2708	no	yes	no	only 16K board avail. for LSI-11 (DEC) and H-11 (Heath) we know about
0	Intel 2716 or 2708	no	yes	no	2708s at reduced rate of ordered with PC board
0	1702A, 5203	no	yes	no	no jumpers on-board (cf. text)
0	2708	yes	yes	no	(cf. text)
0	2708	no	yes	no	(cf. text)
4K	82S126 PROM; 2112 RAM	no	no	yes	power up/restart/jump circuit to any 1K boundary (cf. text)
0	2708	no	yes	no	-
0	1702	no	yes	no	-
0	2716	no[12]	no	yes	CPU-clock board for MP-68
256B	2708, 2716	yes[13]	no	yes	CPU and I/O for CRT, printer, and disk (cf. text)
0	1702A	no	yes	no	-
1K	[14]	no	yes	yes	power on-jump/reset + run; circuit for MWRITE logic (cf. text)
0	[15]	yes	yes	no	(cf. text)
2K	2708	yes	no	yes	CPU/clock board, 6K memory, EPROM programmer (cf. text)
0	2716	no	yes	no	-
2K	2114 RAM, [4]	no	no	yes	I/O, RAM, cassette interface (cf. text)
1K	1702A	no	yes	yes	reset and go function including phantom output disable, wait state
1K	2704, 2708	no	yes	yes	jumper options, jumper selectable addressing

10 uses 1½ I/O ports; **11** not yet available; **12** programmer board (MP-R) avail. @ $44.95; **13** w/ opt'l attachment; **14** 1702A, 2704, 2708, 2716 Intel, 2716 TI, 5204, 6834; **15** 1702A, 2704, 2708, 2716 Intel, 2716 TI, 5204 Nat'l, S5204 AMI, 6834!; **16** the super system univ'l bus: 10 16-pin IC sockets intercon. by flat rib'n cable; easy S100 con'ct; **17** Intel μ dev. sys.; **18** $33-50; **19** $43-60; **20** $89-129; **21** $169-119.



Oliver Advanced Engineering's 2708/2716 EPROM Programmer

Optimal Technology's 2708/2716 EPROM Programmer

Cromemco's 32K Bytesaver™

use of a variety of firmware will enhance SOL's already impressive position in the microcomputer realm.

RDA of Beltsville, Maryland, offers the only 16K board we know of available for the LSI-11 (DEC) and H-11 (Heath). This board, the RMRV-8K, uses the DEC LSI-11 "Q-bus".

S.D. Sales manufactures the ExpandoPROM board, which will soon be offered completely assembled, but now comes in kit form for under $50. If you order 2708s with the board, S. D. Sales reduces their price per PROM by about 25%! One of the important features of the ExpandoPROM board is DIP switch selection. Maximum capacity of this board, excellent for storing large amounts of software for immediate execution, is 32K using Intel 2716s. This board is a significant bargain for hobbyists and small systems users.

Seals Electronics has a 4K ROM board that will work with the weakest power supply of an S100 bus computer. Switch selectable wait states allow even the slowest 1702A to work in the system. There are 0-7 wait states possible. Programming is available at the factory for $3 per EPROM when accompanied by binary-formatted tape. Seals has an excellent reputation for fast service and high quality products.

Smoke Signal Broadcasting offers the P-38 EPROM Memory Board. The P-38, in addition to space for 2708, includes a socket that can accommodate MIKBUG, SWTBUG, or MINIBUG II monitors. Addressing is such that, by switch selection, a 2708 may contain the E000 locations occupied by MIKBUG as well as the FFFF restart locations. SMARTBUG, a 1K monitor program, is available from Smoke Signal Broadcasting on a 2708 for $39.95. It's MIKBUG compatible, but contains several improvements such as I/O through an ACIA, breakpoint capability and single step trace capability.

The POP-1 by SSB interfaces with the company's P-38-1 and P-38-FF EPROM boards. Complete software is provided on cassette for programming. A unique adaptive programming technique allows most 2708s to be programmed in 15 seconds instead of the usual one and a half minutes. A separate self-contained power supply for the programming voltage insures sufficient current capability to program EPROMs from any manufacturer. This unit offers several good features in a small but efficient package. 6800 owners, check these specs!

Solid State Music's MB-9 4K static PROM/RAM board can use a variety of PROM chips: 82S129, 74S287, 74S387, DM8573 and DM8574. The PROM is valid for 5 MHz! RAM or PROM can be mixed in 256 byte increments up to 4K.

Another board by Solid State Music, the 8K/16K EPROM board, uses 2708s, has a selectable wait state circuit and allows switch selectable enabling of only 8K of the board.

Their third PC board, the MC-3 2K/4K EPROM board, also has selectable wait state circuitry and contains select switches for only 2K enable of PROM.

Southwest Technical Products manufactures the MP-A2 processor board, used in their current model of the MP-68. ROM/EPROM capability comes almost free when made part of the processor board. Common buffers are used for both the EPROMs and the CPU. The address of the ROM/EPROM is DIP switch selectable.

SWTP also offers a programmer board, the MP-R, for 2716 EPROMs.. All software necessary to operate the board is supplied with the kit and includes provisions for testing, verifying, and copying 2716 EPROMs. The board consumes about 0.4A during programming and 0.15A while idle. The MP-R sells for $44.95 and is an option for the SWTPC 6800 Computer System.

Space Byte's 8085 CPU board can contain either 3K or 6K of PROM and 256 bytes of RAM. With an optional attachment, this board can program a PROM – either the 2708 or the 2716. The 8085 CPU is primarily a CPU board, but also serves as an I/O board for CRT display, printer and disk system. Extensive software is available.

Szerlip Enterprises offers both a PROM board for S100 bus computers and a programmer. As indicated on the chart, both can be wired for the largest array of ROM chips we've found. The RAM-N-ROM board can operate with smaller increments of ROM address area than used for the full board. The power-on/jump/reset and run will operate with computers with front panels as well as those without. Both the board and the programmer offer versatility in an economical, quality product.

Every company listed in the chart is well known in the industry. They all offer high quality merchandise with extensive service back-up. As you review your needs for a PROM board or PROM programmer, you will find that most specifications are covered in the chart. New manufacturers and new products in this area may have escaped our notice; but if you use the same criteria we've used, you'll find judging unlisted products equally simple and efficient.

By the way, when we hear that you readers have found our charts handy and beneficial, we break out of a stare and perform terpsichorean feats gingerly. What could be more appropriate when discussing PROMS?

PET FILES

Things your user's manual never told you

BY MICHAEL TULLOCH

After six months, several letters, and more than a few telephone calls, I got a pet. Commodore's PET, of course. The thrill of unpacking the machine (opening a box and removing two pieces of foam packing) and getting it up and running (plugging it in and turning it on) was somewhat diminished when I found the instruction book (pamphlet? flyer?). Nine pages of it.

I'd just helped a friend get one of the early TRS-80s running. His instruction book was only slightly more complete than mine. However, 8K PET BASIC isn't as simple as TRS-80 level 1 BASIC. PET uses a file system for data storage. Learning to use this file system without complete documentation was a long process. I hope this article will make it easier for you.

The first few days after my PET arrived, I found eight ways to bomb a program and lock out the keyboard. I also discovered how to use most of the commands and statements. Simple BASIC programs became easy to write. I stayed away from assembler programs because the commands were indecipherable. But I soon found simple programs wouldn't make my PET do what I wanted.

What I really wanted to get running was a stock market records and analysis program requiring input and output of data from tape. Numeric data, at least, and preferably alphanumeric also. PET's instructions contained nary a hint about how to do this trick.

Now, Commodore's BASIC was written by Microsoft and they're responsible for some of the best BASICs around. Certainly, they wouldn't leave out such an important capability?

At this point, I should've visited my friendly computer store for a book or advice. Unfortunately, the nearest computer store is a 5-hour drive away (I'm really out in the boonies). So after a few false starts I dug out an old Digital software book.

Aha! Files! Programs can be stored on files! Data can be stored on files! Come to think, back in "Introduction to Programming" 10 or 12 years ago, they told me big computers use files for input and output. But could my micro work like a "big computer"?

I went back to my nine-page in-

Table I

PET File Commands/Instructions With Manual Explanations

LOAD"NAME"	Loads file NAME from built in tape unit
LOAD"NAME",D	Loads file NAME from device D
SAVE"NAME"	Saves current file or program NAME on built in tape unit
SAVE"NAME",D	Saves current file or program NAME on device D
SAVE"NAME",D,C	Saves current file or program NAME on device D. C specifies EOF or EOT.
CLOSE N	Closes logical file N
GET#D,C	Accepts single character from specified logical file
GET#D,C$	Accepts single string character from logical file
INPUT#D,A	Accepts value A from logical file D
INPUT#D,A$	Accepts string from logical file D
OPEN A	Opens logical file A for read only from built in tape unit
OPEN A,D	Opens logical file A for read only from device D
OPEN A,D,C	Opens logical file A for command C from device D
OPEN A,D,C, "NAME"	Opens logical file A on device D; if device D accepts formatted files, file NAME is positioned for command.
PRINT#D,A	Prints specified value on logical file D
PRINTD,A$	Prints string on logical file D

Table II

Summary of PET File Rules

OPEN A,D,C, "NAME"

OPEN creates a space in internal memory for variables; also positions file "NAME" for INPUT if it is an input file.

A is the file number. Any numeral or variable 1 to 255.

C =0 to INPUT data from an external file to a program.
=1 to PRINT data from a program to a file; can be variable from 0 to1.

D =1 to use internal tape unit.
=2 to use aux tape unit.
=4 - 255 to use an external device with that assignment; can be variable from 1 to 255 except 3.

NAME can be any letters or numbers in quotes or a string variable.

CLOSE A stops further variables from being stored on file A and starts printing the file on tape if it is a PRINT file.

A can be any number or variable 1 to 255 corresponding to an OPENed file.

struction manual. Table I lists the commands and statements I found referring to files.

I'd already figured out how to SAVE programs and LOAD them. Well, a program is a kind of file, but how do you SAVE data? I tried a number of ways to "SAVE" data files – but the list is too long and too embarrassing to print. Anyway, none of them worked.

Next I tried looking at the time share terminal manual at work. A CDC isn't a PET. But I did find out I should OPEN a file before I could do anything with it.

Moving right along, I consulted the computer center personnel at a nearby Air Force base. They suggested that, after opening the file, I had to put data on it. Since a WRITE command didn't exist, perhaps I should PRINT.

Aha (again)! Let's see. OPEN file, then PRINT. That procedure led to the following illuminating exchange:

```
10 X=3
20 OPEN A
30 PRINT#A,X

RUN

?SYNTAX ERROR IN 20
```

Perhaps, since A is used as a variable in other instruction examples, I should use a number:

```
10 X=3
20 OPEN 1
30 PRINT#1,X

RUN

PRESS PLAY ON TAPE # 1
OK

?NOT OUTPUT FILE ERROR IN 30
```

Great! At least the error message is a file error message.

About this time I learned that I had to CLOSE a file before I could OPEN it. No, that's not right – you can't OPEN the same file twice.

The obvious next step was OPEN A, D, C which "opens logical file A for

Number. Number! I hadn't tried using a number for . . . I wasn't impolite, but I didn't finish the brew. Home to the PET.

command C from device D". A bit of intelligent guessing that D for device should be 1 was confirmed. I'd already seen that a letter caused a syntax error. Some experimenting showed that a 3 caused 3 to be printed with no other apparent result. Any number above 3 caused a DEVICE NOT PRESENT ERROR IN 30 message. Using D=2 elicited PRESS PLAY ON TAPE # 2. So, D must be equal to 1.

But what was command C? The only other reference to C in the instruction manual was C specifying EOF or EOT. Not much help. Perhaps C was SAVE. SAVE was listed under the command column. SYNTAX ERROR IN 20. How about WRITE? SYNTAX ERROR IN 20. PRINT? SYNTAX ERROR IN 20. More crafty now – OUT. No error message. Run program:

```
10 X=3
20 OPEN 1,1, OUT
30 PRINT #1, X

RUN

?NOT OUTPUT FILE ERROR IN 30
```

I was back where I started. Then a wild guess OPEN 1, 1, DATA – syntax error.

Hmm –

```
10 X=3
20 OPEN 1,1,ZZ
30 PRINT #1

RUN

?NOT OUTPUT FILE ERROR IN 30
```

I then found that any non-command group of letters could be substituted for ZZ in line 30. This fact should have been the clue, but it didn't sink in just then.

Things stayed this way for a full week. I was beginning to reconcile myself to waiting for the promised full instruction manual. But I hated to admit defeat. Worse yet, if the instruction manual took another six months, I'd probably beat my lovely, otherwise perfect $800 PET to death. Then I'd have to worry about not only the Humane Society, but also my loving spouse.

One Saturday night my TRS-80-owning buddy invited me over to discuss the limitations of his Tiny BASIC, and his plans for all those goodies Tandy was hinting at – and not incidentally to have a brew. He'd received his full, complete instruction manual. It was a good introduction both to the TRS-80 and BASIC as well.

Hope flickered again. Perhaps the TRS-80 manual could give me some insight into data file storage.

Alas, Tiny BASIC takes a very simple approach to data storage. We were ready to concede, a few cans later, that I'd either have to wait or just try every

possible command, letter arrangement and number.

Number. Number! I hadn't tried using a number for C. A was a number. D was a number. Why had I assumed C was a literal command?

I wasn't impolite, but I didn't finish the brew. Home to the PET!

```
10 X=3
20 OPEN 1,1,1
30 PRINT#1,X
```

```
RUN
```

No error message. But also no tape movement. Well, maybe C=2, so I typed it in:

```
CLOSE 1
```

then pushed RETURN. Lo and behold the display printed:

```
PRESS PLAY AND RECORD ON
TAPE UNIT #1
```

Success!

This may not be the most brilliant way to figure out an undocumented language structure, but it proved successful. Further experimentation led to the following discoveries:

In the statement OPEN A, D, C, "NAME"

1) A,D, and C can all be variables
2) A can be any number from 1 to 255
3) D is 1 for internal tape unit; D is 2 for aux tape unit 2; D can be 4 to 255 for external devices
4) C is 0 for INPUT from an external device; C is 1 for PRINT to a file from a program
5) "NAME" can be either a string or any number in quotes

For example, if A$="23", A=3, D=1, C=1 then the statement OPEN A, D, C, A$ creates file 3 in internal memory which can be PRINTed on using the statement PRINT#3,X. X can only be a single variable or string. PRINT#3, x,y,z is not allowed. When the CLOSE 3 statement is executed, a file named "23" (note: not number 23) is saved on tape unit 1.

A caution should be mentioned here. If the tape unit has the play key depressed, no prompting will occur. However, the file will not be saved unless the record key is depressed. The PET monitor only senses the play key position.

Inputting from a data file is straightforward:

```
10 OPEN 1,1,0, "23"
20 INPUT#1,X,Y,Z,A$
```

Line 10 positions the tape to the beginning of file "23". The tape advances as each variable or string is INPUT. Notice that multiple input variables are allowed. Care must be taken to INPUT variables in the same order as they were PRINTed. It's helpful to keep a written file record.

Now that I understand my PET files, I really need that second tape unit, or a floppy, or . . .

My "Introduction to Your New PET Personal Electronic Transactor" came a few weeks later. It's a bit bigger, with 38 pages. But it *still* doesn't explain the file system.

Table 2 summarizes the PET file rules I discovered. I glued this table to my user's manual for easy reference.

Now that you know how to use your PET's files, you can really tackle data management jobs. Business data, stock prices, magazine indices – even bridge scores – can be FILEd away.

Tumult At The Toronto Tournament

The following article appeared in Computer News #153, University of Toronto, and was, itself, a reprint from an article appearing in the October 1977 issue of the Ryerson Computing Centre Newsletter:

"The Toronto II Room of the Hotel Toronto, normally a setting for testimonial suppers or award banquets, saw a gathering of an altogether different sort this August. The occasion was the World Computer Chess Championships, part of the proceedings of the mammoth IFIP (International Federation for Information Processing) Congress 77 which the city of Toronto hosted this year.

"Sixteen chess programs from across the globe participated in the tournament, including *Kaissa*, U.S.S.R. The standard international system of competition was used, in which each contestant played four games. Each win counted for one point, each draw for half a point and each loss for no points, with the winner being decided by its total. Play was distributed over three days – two games on the opening Sunday and one game on each of the two following evenings. Spectator attendance was excellent throughout the competition. On the final evening, the room was packed to such an extent by aficionados and the curious that the management had to close the doors. It was observed that on the occasion there were at least twice as many people in attendance as the spectators that came to watch Bobby Fisher retain his last U.S. Chess Championship.

"In the gallery, the atmosphere was one of amazement and unwavering attention as criticisms and words of praise in a dozen different languages passed back and forth. In my section, I was quickly drawn into conversation with two elderly foreign gentlemen about the progress of a game close to us. They put me to shame with their knowledge of chess trivia and tactics. Upon learning that they were novices as far as computers were concerned, however, I was able to dazzle them with my repertoire of facts, (mostly gathered during research, on an earlier Newsletter article on computer chess).

"The crowd was able to keep track of the progress in the various games by means of eight large, magnetic chess boards raised on a platform at the front of the hall. On the floor in front of them sat the contestants – or rather their teams of agents and authors. Each pair of competing teams occupied one table. In many cases one man from each side faced one another across a chess board, lost in deep thought as if they, not the machines, were making the decisions. Some insisted that this gave them a better perspective, allowing them to more accurately gauge the performance of their respective programs. Each table also held the myriad display screens, terminals and modems which were hooked up by telephones to the individual computers into which the chess programs had been placed for the tournament.

"Much of the computer time for the programs was provided free of charge by various installations and companies

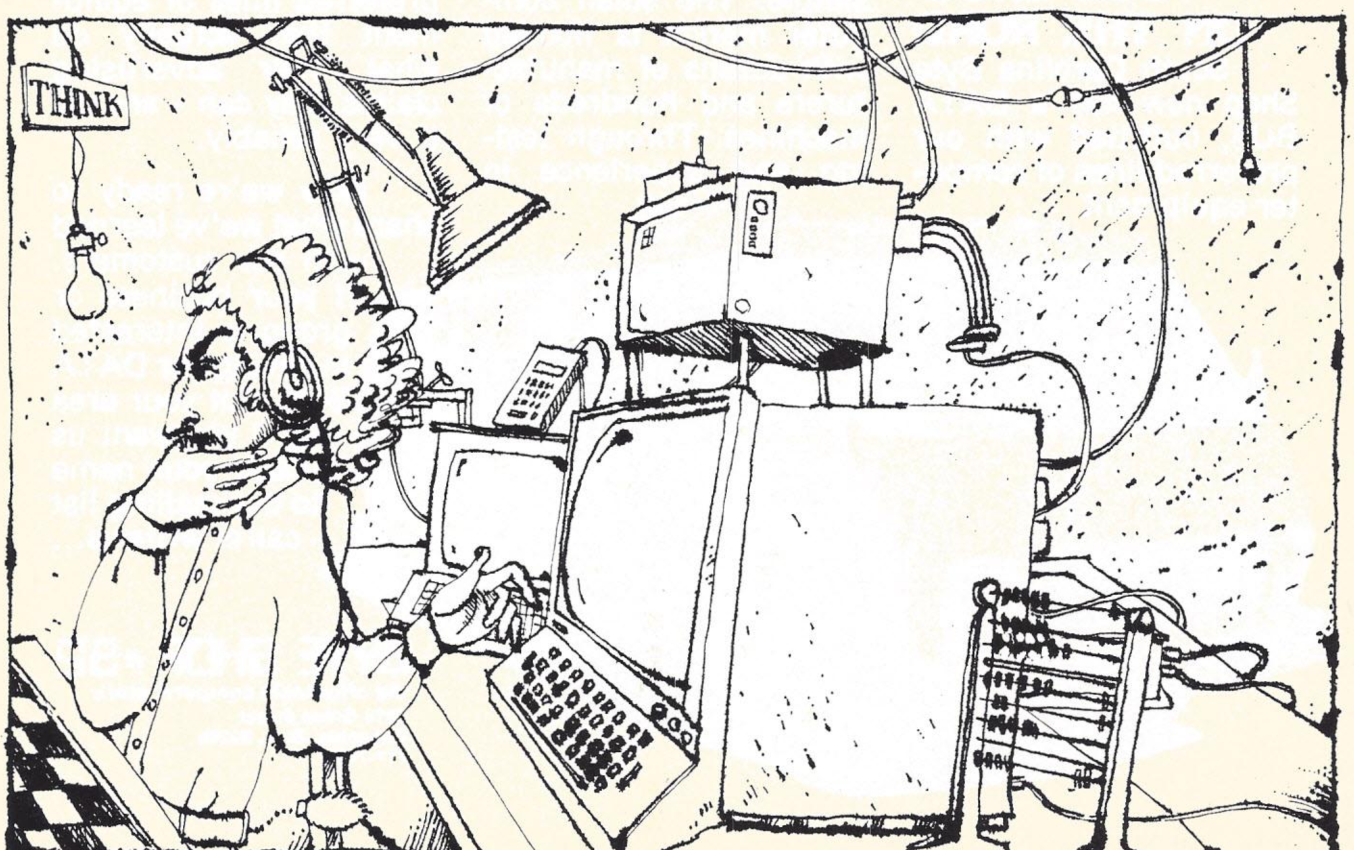


Illustration by Casserine Toussaint

in the region. One representative from a participating computer firm was running back and forth to make telephone calls. The program that his company was hosting already had 95% of an IBM 370-168 dedicated to it, and the competing programmer was hollering for, 'more, more'!" (That program placed next to last in the tournament.)

"The nature of the contestants made silence unnecessary, a fact obvious from the din being generated by the crowd and the media covering the event. The tournament sponsors were therefore able to provide a combination master of ceremonies and commentator.

"Throughout the competition, British chess expert David Levy paraded back and forth along the platform, microphone in hand, providing explanations, comments and predictions about the ongoing games. He was also quick to provide unorthodox criticism of the somewhat unorthodox moves which some programs produced from time to time.

"Long a follower of the growth of computer chess, Mr. Levy told the audience that, despite the obvious flaws that many programs displayed, he was impressed by the quality of play. He went so far as to predict that in ten years, computers would be providing the most superior level of chess competition available in the world.

"The eventual winner of the tournament was *Chess 4.6* from Northwestern University in Illinois, runner-up in the previous champsionship. It was the competition's only undefeated or untied program, finishing one point ahead of its arch-rival, *Kaissa*, from Russia.

"The Russians had problems from the start, losing their opening game to a not particularly strong program. Rumours circulated, attributing the loss to everything from a transient bug in the program to the less than usual style of play produced by the opponent. Whatever the cause, *Kaissa* won the remainder of its games but had to settle for second place. Arguments over which was really the more superior program, *Chess 4.6* or *Kaissa*, were rampant, since the two did not play one another during competition. The matter was settled by an exhibition game where, for the first time, *Chess 4.6* managed to beat the Russians.

"It is impossible to deny the impressive quality of *Chess 4.6*'s play. It is already able to beat 99.5% of all competing players in the U.S. Chess Federation. Throughout the tournament it showed flawless offensive strategy combined with excellent almost impenetrable defence. It has those openings which experts consider to be the best pre-programmed into it. Its middle game is solid, with good scheduling of priorities which prevent the abondoning of a superior position in order to pursue some less desirable goal. Finally, its endgame (traditionally the weak area of play by computers) is methodical, cautious and effective. It also possesses the valuable trait of being able to 'think' during its opponent's timeframe, calculating the most likely move for its competitor and projecting accordingly. Since a time limit for total processing was in effect for the championship, this gave *Chess 4.6* a decided edge.

"One of the highlights of the competition was the appearance of Dr. Mikhail Botvinnik on the final evening of tournament play. As well as being a professor of Electrical Engineering at Moscow University and a pioneer of computer chess, Dr. Botvinnik was the world chess champion for twelve successive years. David Levy jokingly explained the use of the 'French Defence' by *Chess 4.6* in its final game as a tribute to the presence of the old master, since it had always been one of Botvinnik's favorites during his years as an active player."

D. Dowhal
U. of Toronto

The array of giant computers facing each other in the Toronto tournament was ominous. They were like restless armies listening for a bugle call to summon them into battle. One could almost sense the thunder and lightning waiting to erupt when the first skirmish began. The awesome pyrotechnic-display, fortunately, never materialized; but the intellectual giants locked in this great, silent war overwhelmed anyone who pondered for a minute on what was going on in the tournament hall. Accompanying tables give a closer look at these mechanical gladiators that waged war in Toronto.

Technical Details on Participants
Second World Computer Championship

PROGRAM	PROGRAMMING LANGUAGE	SIZE OF OPENING BOOK	PROGRAM SIZE (MEMORY)	AVERAGE NUMBER OF POSITIONS EXAMINED PER MOVE
1. Kaissa	Assembly	10,000 positions	250K	90,000
2. Chess 4.6	Assembly	5,600 positions	7.5 K words (60 - bit) + Ext. Core	400,000
3. Blitz V	FORTRAN IV	5,000 positions	24K words (32 - bit)	500
4. Master	PL/I	450 variations	170K	100,000
5. Tell	Algol 60		15K words (36 - bit)	
6. Duchess	PL/I and Assembly	3,000 positions	300K	100,000
7. Belle	"C"	10,000 positions	8K words (16 - bit)	30,000
8. Wita	Algol W	9,000 positions	350K	250

9. Ostrich	Assembly	no book	20K words (16 - bit)	10,000
10. Dark Horse	Fortran IV	no book	20K - words (60 - bit)	12,000
11. BCP	Fortran and Assembly	1,000 positions	24K words (50 - bit)	1,000 per second
12. BS '66 '76	Fortran	1,000 positions	200K	150
13. Elsa	Assembly	500 positions	100K words (48 - bit)	
14. Chaos	Fortran	7,500 positions	3 Megabytes	30,000
15. Black Knight	Fortran	70,000 positions	30K words (36 - bit)	7,500
16. Chute 1.2	BPL (extended XPL)	45 variations	250K	900

Background on the 16 Participants

PROGRAM	AUTHORS	AFFILIATION	COMPUTER	LOCATION OF COMPUTER
1. Kaissa	Dr. M.V. Donskov Dr. V. Arlazarov	Institute for System Studies Moscow, USSR	IBM 370/168	Canada Systems Group Toronto, Ontario, Canada
2. Chess 4.6	David Slate Larry Atkin	Northwestern University Evanston, Illinois, USA	CDC CYBER 176	Arden Mills, Minnesota, USA
3. Blitz V	Robert Hyatt	University of Southern Mississippi Hattiesburg, Mississippi, USA	SIGMA 9	University of Southern Mississippi Hattiesburg, Mississippi
4. Master	J.A. Birmingham Peter Kent	Rutherford Lab and AERE Harwell, Oxfordshire, U.K.	IBM 370/168	AERE Harwell, U.K. (via I.P. Sharp Assoc.)
5. Tell	Johann Joss	Eidgenossische Technische Hochschule, Zurich, Switzerland	DEC K110	Dataline Systems Toronto, Ontario, Canada
6. Duchess	Tom Truscott Bruce Wright Eric Jansen	Duke University Durham, N.C., USA	IBM 370/165	Triangle Universities Computer Centre Triangle Park, N.C., USA
7. Belle	Ken Thompson Joe Condon	Bell Telephone Labs Murray Hill, New Jersey, USA	PDP-11	Bell Telephone Labs Murray Hill, New Jersey, USA
8. Wita	Tony Marsland	University of Alberta Edmonton, Alberta, Canada	AMDAHL 470 V/6	University of Alberta Edmonton, Alberta, Canada
9. Ostrich	Monty Newborn George Arnold	McGill University Montreal, Quebec, Canada	SUPERNOVA	At tournament site (via Data General)
10. Dark Horse	Ulf Rathsman	Telefon AB LM Ericsson Stockholm, Sweden	CDC 6600	Multiple Access Computer Group Toronto, Ontario, Canada
11. BCP	Don Beal	Queen Mary College London, England	CDC 6400	McMaster University Hamilton, Ontario, Canada
12. BS '66 '76	Barend Swets	Private entry Tilburg, Netherlands	IBM 370/168	Datacrown Limited Toronto, Ontario, Canada
13. Elsa	Ludwig Zagler	Technischen Universitat Munchen Munich, West Germany	Telefunken TR 440	Techn. Univ. Munich Munich, West Germany (via I.P. Sharp Assoc.)
14. Chaos	Mike Alexander T. McBride Fred Swartz Bill Toikka Bic Berman Joe Winograd	University of Michigan Ann Arbor, Michigan, USA	AMDAHL 470 V/8	Amdahl Corporation Sunnyvale, California, USA
15. Black Knight	Ken Sogge Fred Prouse Gary Maltzen Lonny Lebahn Elliot Adams	Sperry Univac St. Paul, Minnesota, USA	UNIVAC 1110	Sperry Univac Roseville St. Paul, Minnesota, USA
16. Chute 1.2	Mike Valenti Zvonko Vranesic	University of Toronto Toronto, Ontario, Canada	AMDAHL 470 V/6	Industrial Life – Montreal, Quebec, Canada

Technical Details on Participants Second World Computer Championship

1. **Kaissa**
Dr. M.V. Donskoy
Dr. V. Arlazarov
Inst. of Control Sciences
Profsojvznaya 81
Moscow, 117485
U.S.S.R.

2. **Chess 4.6**
Mr. David Slate
Vogelback Computer Ctr.
2129 Sheridan Road
Evanston, IL 60201
U.S.A.

3. **Blitz V**
Dr. Robert Hyatt
Box 286
Southern Station
Hattiesburg, MS 39401
U.S.A.

4. **Master**
J.A. Birmingham
Computer Science
and Systems Division
Aere Harwell, Oxfordshire
OX11 ORA
England

5. **Tell**
Johann Joss
Eidgenossiche
Technische Hochschule
Seminar fur
Angewandte Mathematik
Eth-Zentrum
CH-8092 Zurich
Switzerland

6. **Duchess**
Tom Truscott
Dept. of Computer Sci.
Duke University
Durham, NC 27706
U.S.A.

7. **Belle**
Ken Thompson
Bell Laboratories
600 Mountain Avenue
Murray Hill, NJ 07974
U.S.A.

8. **Wita**
Dr. T.A. Marsland
Dept. of Computer Sci.
Univ. of Alabama
Edmonton,
Alberta,
Canada

9. **Ostrich**
Dr. Monroe Newborn
Sch. of Computer Sci.
McGill University
P.O. Box 6070, Station A
Montreal, Quebec
H3C 3G1

10. **Dark Horse**
Ulf Rathsman
Heleneborgsgatan 7
4 tr.t.h.
11731 Stockholm
Sweden

11. **BCP**
Don F. Beal
Dept. of Computer
Sci. and Stat.
Queen Mary College
University of London
Mile End Road
London, E1 4NS
England

12. **BS.66/76**
B. Swets
Conservatorium Laan 15
Tilburg, Netherlands

13. **Daja**
Dr. Ludwig Zagler
Institut fur Informatik
der Technischen
Universitat Munchen
D-8000 Munchen 2
Postfach 20 24 20
West Germany

14. **Chaos**
Fred Swartz
Computing Center
University of Michigan
1075 Beal Avenue
Ann Arbour, MI 48108
U.S.A.

15. **Black Knight**
Ken Sogge
Sperry Univac Roseville
M.S. 4873
P.O. Box 3942
St. Paul, MN 55165
U.S.A.

16. **Chute 1.2**
Mike Valenti
774 Spadina Road
Toronto, Ontario
Canada

17. **Pioneer**
(unable to participate)
Dr. M. Botvinnik
VNIIE
16 Bersenevskaja nab.
Moscow J-72
U.S.S.R.

18. **Orwell**
(unable to participate)
Thomas Nitsche
Briennerstr. 49
800 Munchen 2
Germany

19. **Tournament Director**
David Levy
104 Hamilton Terrace
London, NW8 9UP
England

20. **Tournament Coordinator**
Prof. D. Michie
Machine Intelligence
Research Unit
Univ. of Edinburgh
Hope Park Square
Meadow Lane
Edinburgh EH8 9NW
England

21. **Tournament Coordinator**
Dr. Zvonko Vranesic
Dept. of Elec. Engineering
Univ. of Toronto
Stanford Fleming Bldg.,
Toronto, Ontario
Canada MSS 1A1

The Eight Games of the First Toronto Round

White – **Ostrich** Black – **Wita**

1. P-K4	P-Q B4
2. N-KB3	P-Q3
3. B-B4	N/KN1-KB3
4. NP-QB3	P-K3
5. 0-0	B-K2
6. P-Q3	B-Q2
7. B-K3	N/QN1-QB3
8. B-QN5	0-0
9. N-KN5	P-KR3
10. BxN	BxB/QB3
11. N-R3	K-KR2
12. Q-B3	N-Q2
13. KR-QN1	Q-QR4
14. Q-KN4	B-KB3
15. B-Q2	B-Q5
16. Q-R4	Q-Q1
17. QxQ	Q/QR1xQ
18. N-K2	P-KB4
19. NxB	P/QB4xN
20. N-B4	R/Q1-K1
21. N-K2	P-K4
22. PxP	R/KB1xP/KB4
23. P-KB4	N-QB4
24. B-N4	P-KN3
25. PxP	R/KB4xP/K4
26. NxP	B-Q4
27. P-B4	NxP/Q6
28. BxP	R/K4-K5
29. N-N5	BxP/QB5
30. NxP	R/K1-QR4
31. P-QN3	B-QR3
32. P-QN4	RxN
33. P-N5	P-N3
34. PxB	R/QR2xP/QR3
35. B-B7	N-QN5
36. P-QR3	N-QB7
37. R-R2	R/K5-K8ch
38. RxRch	Nxr/K8
39. K-B2	N-Q6ch
40. K-B3	N-QN5
41. R-N2	N-Q4
42. B-Q6	P-QN4
43. B-B5	R-KB3ch
44. K-K4	N-QB2
45. R-KB2	N-K1
46. RxR	NxRch
47. K-Q3	K-KN2
48. B-Q4	K-KB2
49. BxN	KxB
50. K-Q4	K-K2
51. K-B5	K-Q1
52. KxP	K-QB2
53. P-QR4	P-KN4
54. P-R5	K-QN1
55. K-N6	K-QB1
56. P-R6	K-QM1
57. P-N4	K-QR1
58. P-R7	P-KR4
59. PxP	P-KN5
60. P-R6	P-KN6
61. P-R7	P-KN6
62. Black Resigns	

White – **Chaos** Black – **BS '66 '76**

1. P-Q4	P-QB4
2. P-QB4	P-K4
3. P/Q4xP	P-Q5
4. N-KB3	N-QB3
5. QN1-Q2	B-K3
6. P-KN3	P-KB4
7. B-N2	P-Q6
8. 0-0	R-N1
9. PxP	QxP
10. Q-N3	Q-Q1
11. N-K4	N-Q5
12. NxN	QxN
13. B-K3	R-Q1
14. BxQ	RxB
15. QxP	BxP
16. R/KB1	B-Q6
17. Q-N8ch	K-Q2
18. QxB	P-QB4
19. RxB	K-B2
20. QxP/QB5ch	K-N1
21. Q-KB8ch	K-B2
22. RxR	N-K2
23. QxN ch	K-N3
24. Q-K4ch	

Black Time Forfeits

White – **Belle** Black – **Black Knight**

1. P-K4	P-QB4
2. N-KB3	N-QB3
3. P-Q4	PxP
4. NxP	K-B3
5. N-QB3	P-Q3
6. B-QN5	B-Q2
7. 0-0	NxN
8. QxN	BxB
9. NxB	P-K4
10. Q-R4	N-Q2
11. R-Q1	P-Q3
12. NxPch	BxN
13. RxB	Q-B2
14. R-Q3	P-QN4
15. Q-N3	0-0
16. B-N5	N-B4
17. Q-B3	R-R2
18. B-Q8	RxB
19. QxN	RxR
20. QxQ	RxQ
21. PxR	R-B7
22. P-QN3	P-QR4
23. P-B3	PB3
24. P-QR4	P-N5
25. P-B4	R-B6
26. PxP	PxP
27. R-N1	RxQP
28. K-B2	P-N4
29. R-N2	P-R4
30. K-K2	R-QB6
31. R-N1	R-B7ch
32. K-B1	R-B7ch
33. R-B1	R-N7
34. R-B8ch	K-N2
35. R-B7ch	K-B3
36. R-B6ch	K-B2
37. R-B5	R-N8ch
38. K-B2	Adjudicated a Draw

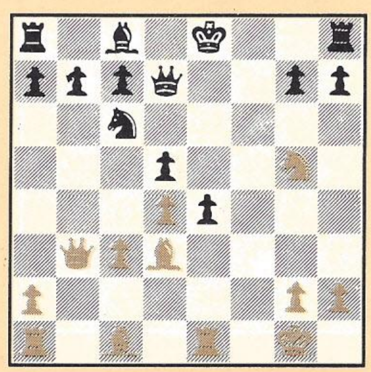
White – **Chute 1.2** Black – **Master**

1. P-P4	P-Q3
2. N-QB3	P-KN3
3. B-B4	B-K6
4. BxB	PxB
5. N-B3	N-QB6
6. Q-K2	B-N2
7. 0-0	Q-Q2
8. P-QR3	0-0-0
9. P-QN3	P-Q4
10. PxP	PxP
11. R-R2	P-Q5
12. N-Q1	P-Q6
13. PxP	Q-Q4
14. B-N2	P-K4
15. R-K1	QxNP
16. N-B3	N-Q5
17. Q-K4	NxNch
18. QxN	N-R3
19. Q-K3	R-Q5
20. B-R1	N-N4
21. Q-R3	K-N1
22. R-K4	P-KR4
23. Q-N3	Q-K3
24. P-R3	N-R3
25. P-KB4	N-B5
26. Q-B3	N-Q3
27. RxR	PxR
28. N-Q5	P-B3
29. N-N4	Q-K8ch
30. Q-B1	R-K1
31. QxQ	RxQch
32. K-B2	R-KR8
33. P-QR4	K-B2
34. N-B2	P-B4
35. N-K1	N-B4
36. R-B2	K-Q3
37. R-N2	P-N3
38. R-N1	P-R5
39. R-B1	N-N6
40. K-B3	R-B1ch
41. K-N4	N-K7
42. R-N1	B-B3
43. P-N3	RPxP
44. P-R4	P-N7
45. BxP	

Adjudicated win for black

White – **Chess 4.6** Black – **BCP**

1. P-K4	P-K4
2. N-KB3	N-QB3
3. P-Q4	PxP
4. P-B3	Q-K2
5. PxP	QxPch
6. B-K2	P-Q4
7. N-B3	B-QN5
8. 0-0	BxN
9. B-Q3	Q-K2
10. PxB	N-B3
11. Q-N3	N-K5
12. R-K1	Q-K3
13. N-N5	Q-Q2
14. P-B3	P-B4
15. PxN	BPxP



Black is now a dead duck as the World's Champ, **Chess 4.6** launches its attack. See Fig 2.

16. BxP	PxB
17. RxPch	N-K2
18. Q-B7ch	K-Q1
19. QxP	Q-K1
20. N-B7ch	K-Q2
21. NxR	K-Q3
22. RxN	QxR
23. B-R3ch	K-B3
24. QxQ	P-KR4
25. Q-B5ch	K-Q2
26. R-K1	P-QR3
27. Q-Q5 Mate (and still champ!)	

White – **Duchess** Black – **Kaissa**

1.	P-K4	P-Q4
2.	PxP	N-KB3
3.	P-Q4	NxP
4.	N-KB3	P-KN3
5.	B-K2	B-N2
6.	P-B4	N-N3
7.	N-B3	0-0
8.	B-K3	B-N5
9.	P-B5	N-Q4
10.	0-0	P-K3
11.	Q-N3	P-N3
12.	NxN	PxN
13.	B-KN5	Q-Q2
14.	P-KR3	B-B4
15.	Q-B3	R-K1
16.	KR-K1	B-K5
17.	N-Q2	Q-B4
18.	B-K3	Q-K3
19.	NxB	PxP
20.	PxP	Q BPxP
21.	KR-QB1	N-Q2
22.	B-N4	Q-Q4
23.	Q-B6	N-B3
24.	B-K2	QR-Q1
25.	Q-R4	R-K2
26.	B-N5	Q-B4
27.	R-B2	N-Q4
28.	QR-B1	B-B3
29.	Q-N3	P-QR4
30.	P-N4	Q-K3
31.	R-B6	P-R5
32.	QxP	R-Q3
33.	RxR	QxR
34.	Q-R8ch	R-K1
35.	QxRch	K-N7
36.	P-N5	B-Q1
37.	B-QB4	Q-K2
38.	QxQ	NxQ
39.	B-B4	N-B4
40.	B-Q5	K-B1
41.	R-B8	K-K2
42.	R-B4	N-N2
43.	BxP	N-K3
44.	B-K3	N-B2
45.	P-Q5	N-N4
46.	B-B3	K-Q2
47.	P-QR4	N-Q3
48.	R-B6	N-B4

White Wins

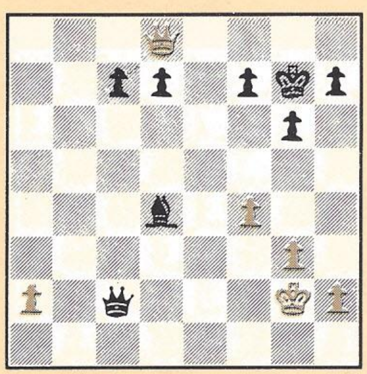
White – **Elsa** Black – **Blitz V**

1.	P-Q4	P-Q4
2.	P-QB4	P-K3
3.	N-QB3	N-KB3
4.	B-N5	B-K2
5.	N-B3	0-0
6.	Q-N3	PxP
7.	QxBP	QN-Q2
8.	0-0-0	N-N3
9.	Q-Q3	B-Q2
10.	N-K5	R-B1
11.	Q-KB3	KN-Q4
12.	BxB	QxB
13.	NxB	QxN
14.	P-K4	NxN
15.	QxN	Q-R5
16.	P-QR3	KR-Q1
17.	Q-B5	R-Q3
18.	B-N5	Q-R4
19.	P-KB4	P-QR3
20.	B-K8	QxQch
21.	PxQ	RxRch
22.	RxR	RxB
23.	PxN	PxP
24.	P-QN4	P-KR3
25.	K-N2	R-K2
26.	R-Q6	P-QN4
27.	P-KN4	R-B2
28.	K-N3	K-B1
29.	P-K5	R-B5
30.	R-Q8ch	K-K2
31.	R-QN8	R-B2
32.	P-KR4	P-KN3
33.	P-N5	P-KR4
34.	R-QR8	K-Q2
35.	R-QN8	K-B3
36.	R-Q8	R-Q2
37.	RxR	KxR
38.	K-B3	K-B3
39.	K-Q4	P-N3
40.	K-K4	P-R4
41.	K-Q4	P-R5
42.	K-K4	K-Q2
43.	K-Q4	K-K2
44.	K-K4	K-B1
45.	K-Q4	K-K1

Adjudicated a draw.

White – **Tell** Black – **Dark Horse**

An unusual game occured when **Tell** battled **Dark Horse**. The game started off in very lively fashion, when Black decided to employ the two Knights Defense. By the end of the first 11 moves, five major pieces had departed the board. By the end of the 30th move, the board was pretty well swept clean of chessman. Black, at this point has a decided advantage.



1.	P-K4	N-KB3
2.	B-QB3	N-QB3
3.	P-Q4	P-Q3
4.	B-QN5	B-Q2
5.	B-KN5	P-K4
6.	BxKN	QxB
7.	N-Q5	Q-Q1
8.	N-B3	PxP
9.	KNxP	B-K2
10.	BxN	BxB
11.	NxQB	PxN
12.	N-N4	Q-Q2
13.	Q-Q4	B-B3
14.	Q-K3	BxP
15.	R-QB1	B-B4
16.	N-R6	0-0
17.	R-N7	KR-QB1
18.	RxRP	Q-N5
19.	0-0	RxR
20.	QxR	QxKP
21.	Q-N7	R-KB1
22.	QxBP	B-K4
23.	R-Q1	Q-K7
24.	R-QB1	QxN
25.	R-N8	P-N3
26.	RxR	KxR
27.	P-N3	Q-B4
28.	Q-Q8ch	K-N2
29.	P-B4	B-Q5ch
30.	K-N2	Q-QB7ch
31.	K-B3	Q-B7ch
32.	K-N4	QxKRP
33.	Q-R5	P-KB4
34.	QxP	PxQ

With this suicide by White's Queen, White was hopeless and his King now began to squirm.

35.	KxP	QxP
36.	K-K4	P-B4
37.	P-R4	Q-N5ch
38.	K-Q3	Q-B6ch
39.	K-B4	QxP
40.	P-R5	QxKB8ch
41.	K-Q5	B-K4
42.	P-R6	Q-B6ch
43.	K-B4	Q-K7ch
44.	K-Q5	QxP
45.	K-K4	Q-K7ch
46.	K-Q5	Q-B6ch
47.	K-B4	B-Q5
48.	K-N5	Q-N2ch
49.	K-B4	P-R3
50.	K-Q3	Q-N6
51.	K-B4	B-K4
52.	K-Q4	

At this point Black was adjudicated the winner, giving respite to an exhausted White King.

8080 Chess, the only microcomputer to compete in the Seattle tournament won its first game in Round 1 when it was matched against *Ostrich*. The win was the result of a time forfeit on the part of *Ostrich*. *8080 Chess* did not fare as well in Round 2 losing to *Chaos* in 16 moves by forfeit because it was unable to make a move within time limits. Nevertheless, *8080 Chess*, by winning a victory in Round 1, astounded the viewers. It had beaten Data General's Supernova computer that had a program size of 20K words (16-bits) and an ability to examine 10,000 positions per move. Specifications of all the participants are listed in the following table:

Chess 4.6 from Northwestern University. Computer: CDC Cyber 176. Memory: 7.5K Words (60-bit). Examines 400,000 positions per move.

Duchess from Duke University. Computer: IBM 370/165. Memory: 300K. Examines 100,000 positions per move.

Chaos, University of Michigan. Computer: Amdahl 470 V/8. Memory: 8 megabytes. Examines 30,000 positions per move.

Wita, University of Alberta. Computer: Amdahl 470 V/6. Memory: 350K. Examines 250 positions per move.

Ostrich, McGill University. Computer: Data General Supernova. Memory: 20K words (16-bit). Examines 10,000 positions per move.

Blitz V, University of Southern Mississippi. Computer: Xerox Sigma 9. Memory: 24K. Examines 5,000 positions per move.

Tyro, University of Southern California. Computer: PDP 10 KL. Memory: 270K. Examines 10,000 positions per move.

Chute, University of Toronto. Computer: Amdahl 470 V/6. Memory: 250K. Examines 900 positions per move.

Xenarbor, Control Data. Computer: IBM 370/158. Memory: 130K. Examines 10,000 positions per move.

Brute Force, University of Manitoba. Computer: IBM 370/168. Memory: 25K. Examines 125,000 positions per move.

8080 Chess, Processors Technology. Computer: Intel 8080 Microprocessor. Memory: 14K. Positions examined: (not available).

The Seattle tournament was run in the four-round Swiss format and lasted three days. Rules of play between computers were identical to regular human tournament play. Games were played at a speed of 40 moves per player for the first two hours and then 10 moves every 30 minutes thereafter. The tournament director had the right to adjudicate a game after four and one-half hours total elapsed time. Machine failures permitted the programmers a 30-minute time out. The clock was restarted in 30 minutes and if it could not move then, a forfeit was declared. Participants were granted permission to stop the clock up to three times during the game, but the total time-outs could not exceed 45 minutes during a game. There was no manual adjustment of program parameters during the course of a game. In the case of failures, the program parameters were reset to original settings if it was at all possible. Information regarding castling status, en passant status, etc., could be typed in after a failure. If at any time during the course of a game, the computer asked for time remaining on either his or his opponenet's clock, this information was provided. However, the computer had to initiate this request for information. At the end of each game, teams were required to turn in game listings.

ACM Chess . . . Round #2

White: **8080 Chess** Black: **Chaos**

1.	P-K4	P-QB4		6.	N-KB3	B-B4	
2.	P-Q4	PxP		7.	B-N5	Q-N3	
3.	QxP	N-QB3		8.	BxN	BxPch	
4.	Q-Q5	N-B3		9.	K-Q2	QxP	
5.	Q-Q3	P-K4		10.	BxP	R-KN1	
				11.	Q-B3	QxQch	
				12.	KxQ	RxB	
				13.	N-Q2	P-Q3	
				14.	N-B4	B-QB4	
				15.	P-KR4	B-K3	
				16.	N-N2	R-N5	

White lost – Time Forfeit

White: **Wita** Black: **Tyro**

1.	P-Q4	P-Q4
2.	P-QB4	P-K3
3.	P-K4	PxKP
4.	N-QB3	B-N5
5.	P-QR3	BxNch
6.	PxB	H-KB3
7.	R-N1	0-0
8.	B-K2	P-QN3
9.	B-B4	P-B3
10.	N-R3	Q-K2
11.	Q-R4	KN-Q2
12.	0-0	B-R3
13.	BxN	NxB
14.	KR-Q1	P-KB4
15.	K-R1	P-K4
16.	PxP	QxKP
17.	Q-N3	K-R1
18.	N-N5	Q-B3
19.	N-R3	R-Q1
20.	RxR	QxR
21.	N-B4	Q-R5
22.	P-N3	Q-B3
23.	R-Q1	B-B1
24.	N-R5	Q-K2
25.	P-KR3	P-N3
26.	P-B5	PxN
27.	PxP	PxP
28.	P-KB4	P-N4
29.	BxRP	B-K3
30.	Q-N4	QxQ
31.	BPxQ	B-Q4
32.	P-N4	PxP
33.	K-R2	PxP
34.	P-B5	RxP
35.	B-N4	R-R7ch
36.	KxP	N-Q2
37.	B-R5	N-B3
38.	K-R4	R-R7ch
39.	K-N5	NxB
40.	P-B6	K-N1
41.	R-KN1	K-B2
42.	R-QR1	NxP
43.	K-B5	R-KB7ch
44.	K-N5	P-R4
45.	K-R6	R-KN7
46.	R-R7ch	K-K3
47.	R-R1	P-K6
48.	R-R6	K-B4
49.	R-R2	R-N3mate

White: **Chute** Black: **Blitz**

1.	P-K4	P-K4
2.	N-KB3	N-KB3
3.	NxP	P-Q3
4.	N-KB3	NxP
5.	Q-K2	Q-K2
6.	P-Q3	N-KB3
7.	B-N5	QN-Q2
8.	QxQch	BxQ
9.	B-K2	O-O
10.	O-O	P-KR3
11.	B-Q2	N-N3
12.	B-R5	B-K3
13.	N-B3	KR-K1
14.	BxN	RPxB
15.	N-Q4	B-Q4
16.	NxB	NxN
17.	B-B3	P-QB3
18.	BxN	PxB
19.	N-B5	QR-B1
20.	QR-K1	B-B1
21.	RxR	RxR
22.	P-Q4	R-K7
23.	N-K3	R-Q7
24.	NxP	RxBP
25.	R-N1	R-Q7
26.	NxP	RxQP
27.	N-Q7	R-Q7
28.	K-B1	P-Q4
29.	K-K1	R-B7
30.	NxB	KxN
31.	R-Q1	RxNP
32.	P-QR4	R-R7
33.	RxP	RxRP
34.	R-Q8ch	K-K2
35.	R-Q3	P-QN4
36.	R-K3ch	K-Q3
37.	R-Q3ch	K-B3
38.	R-B3ch	K-N3
39.	R-B1	R-K5ch
40.	K-Q2	P-N5
41.	P-KB3	R-K4
42.	R-B4	R-QN4
43.	P-N3	P-N6
44.	R-B8	K-N2
45.	R-B8	P-N7
46.	RxPch	K-N3
47.	RxP	P-N8=Q
48.	R-Q7	Q-R7ch
49.	K-Q3	QxP
50.	Resigns	

White: **Duchess** Black: **Black Knight**

Morra Gambit

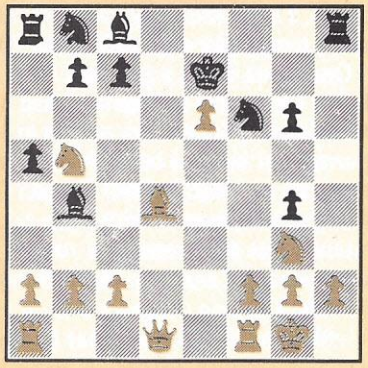
1.	P-K4	P-QB4
2.	P-Q4	PxP
3.	P-QB3	PxP
4.	NxP	N-QB3
5.	N-B3	P-Q3
6.	B-QB4	P-K3
7.	O-O	N-B3
8.	Q-K2	B-K2
9.	R-Q1	P-K4
10.	B-K3	N-KN5
11.	B-Q2	B-R5
12.	B-K1	B-K2
13.	B-Q2	B-K3
14.	BxB	PxB
15.	N-KN5	BxN
16.	QxN	BxB
17.	QxNP	R-KB1
18.	QxNP	Q-N3
19.	QxR8ch	K-K2
20.	QxR8ch	K-Q
21.	RxB	Q-B2
22.	QR-Q1	N-Q5
23.	P-B4	Q-KN2
24.	R-Q3	P-KR4
25.	K-R1	P-R3
26.	PxP	PxP
27.	R-KB1ch	K-K1
28.	R-N3	Q-K2
29.	R-N8ch	K-Q2
30.	R-QR8	K-B3
31.	RxPch	N-K2
32.	R-R4	K-B3
33.	N-R5	Q-N5
34.	RxP	QxP
35.	R-QB5ch	K-Q3
36.	P-K5ch	K-K2
37.	R-QN1	Q-KB7
38.	R-B7ch	K-Q1
39.	R-KR7	N-B3
40.	R-B8ch	K-B2
41.	N-N5ch	K-Q2
42.	R-R7ch	K-Q1
43.	N-Q6	Q-B1
44.	K-N1	P-R5
45.	R(1)-QN7	N-K2
46.	R-N8ch	K-B2
47.	RxQ	P-R6
48.	RxNch	K-B3
49.	PxP	K-Q4
50.	N-B7	K-Q5
51.	RxP	K-Q4
52.	R-Q6ch	K-K5
53.	R-B8	K-B6
54.	R-B4	K-K7
55.	R-K4ch	K-B6
56.	N-N5mate	

White: **Xenarbor** Black: **Chess 4.6**

1.	P-Q4	N-KB3
2.	P-QB4	P-QB4
3.	P-Q5	P-K3
4.	N-KB3	PxP
5.	PxP	P-Q3
6.	P-K3	B-KB4
7.	N-B3	QN-Q2
8.	P-KN3	N-K4
9.	NxN	PxN
10.	Q-N3	Q-N3
11.	QxQ	PxQ
12.	B-N2	B-Q6
13.	P-K4	B-Q3
14.	B-K3	O-O
15.	O-O-O	B-QB5
16.	P-N3	B-QR3
17.	N-R4	B-N4
18.	NxP	R-R3
19.	P-QR4	B-K7
20.	R-Q2	RxN
21.	RxB	RxP
22.	R-B2	R-QB1
23.	B-R3	R-QR1
24.	R-QB4	P-QN4
25.	R-B2	RxP
26.	B-N2	P-B5
27.	P-R4	P-B6
28.	R-K2	NxKP
29.	BxN	RxB
30.	R-QR2	R-QR6
31.	RxR	BxRch
32.	K-B2	B-N5
33.	R-QR1	P-B4
34.	R-R8ch	K-B2
35.	R-QN8	P-B5
36.	K-Q3	PxB
37.	PxP	R-QB5
38.	RxP	P-QB7
39.	KxR	B-K2
40.	P-K4	P-B8=Qch
41.	K-Q3	Q-KB8ch
42.	K-K3	QxR
43.	P-KN4	BxP
44.	K-Q2	Q-B5
45.	P-Q6	QxP
46.	P-N5	BxP
47.	K-B3	P-R4
48.	K-N3	P-R5
49.	K-R2	Q-QN5
50.	K-R1	B-B8
51.	K-R2	Q-N7mate

White: Ostrich Black: **Brute Force**

1. P-K4	P-Q4
2. PxP	QxP
3. N-QB3	Q-K4ch
4. KN-K2	P-QR4
5. P-Q4	Q-KB4
6. B-K3	P-KN4
7. N-N3	Q-N3
8. B-Q3	P-KB4
9. O-O	N-KB3
10. N-N5	K-Q1
11. P-Q5	P-N5
12. B-Q4	P-K3
13. PxP	B-N5
14. BxP	K-K2
15. BxQ	PxB



The loss of the Queen mortally wounds **Brute Force** and **Ostrich** lunges in for the kill. (See Fig 3)

16. BxNch	KxB
17. Q-Q4ch	K-K2
18. Q-N7ch	KxP
19. NxPch	K-Q3
20. NxR	R-K1
21. Q-QB7ch	K-K3
22. KR-K1ch	BxR
23. RxBch	K-B3
24. RxR	B-Q2
25. Q-K5ch	K-B2
26. Q-K7mate	

News Items and Vignettes

..... In looking over the letters received from the readers, we find that they request certain information from other readers. If you can be of help, and if you will send in such help, we will begin an interchange of information:

1. Wanted: General information in the chess field.
2. Wanted: Information on writing programs.
3. Wanted: News for microcomputer chess players.
4. Wanted: News for large-computer users.
5. Wanted: a demonstration of computer chess in action in the magazine. (Perhaps set up a middle game and solicit two programs to fight it out in public view.)
6. Wanted: Solutions to reader problems.
7. Wanted: A full computer-chess program to study.
8. Wanted: Rating systems for programs.
9. Wanted: Information on reference material.
10. Wanted: More emphasis on chess playing than on computer function.
11. Wanted: Information on computer-chess activities in foreign countries.
12. Wanted: Publication of dialogue between top computer programmers.
13. Wanted: End-game problems to be solved by the computer.
14. Wanted: Explanation of an actual chess program and logic that will allow readers to follow the problem.
15. Wanted: In-depth review of computer-chess books done by chess masters.

If any of our readers have information they would like to share with their fellow hobbyists, regarding answers to above questions, please let us hear from you. In connection with above questions, *Doug Penrod*'s Newsletter #2 has much information that could be useful. If you do not already have this issue, please send $1.00 to Personal Computing Magazine (1050 Commonwealth Avenue, Boston, MA 02215) and we will send one out to you. We also have some extra copies of Doug's Newsletter #1 and you can have that one also at $1.00 a copy. Following is an example of the type of information that the newsletter contains:

"A reasonably comprehensive lookahead in a search program requires many CPU operations as well as considerable memory. It follows that a successful program should be written in assembly language to enhance the speed of execution or at least in a higher-level language for which a good compiler is available, such as Fortran. Writing in a language which uses an interpreter will generally increase the execution time for a deep search beyond one's patience or resources. A second implication of this line of reasoning is that individuals who wish to write a chess program on a personal computer should make every effort to develop extremely efficient code and should anticipate a need for at least 16K of RAM." From a letter by *Peter W. Frey*, Northwestern University, 2021 Sheridan Rd., Evanston, IL 60201.

Challenge of the game

..... From an item in the newsletter "Checkpoint", published by Canada Systems Group, Mississauga, Ontario, L5K 1B1:

The International Federation for Information Processing (IFIP) is a multinational federation of professional technical societies concerned with the science and technology of information processing. The Canadian member society of IFIP is the Canadian Information Processing Society (CIPS).

Since 1959 IFIP has conducted triennial meetings of professional people in computer and computer related industries where the state of the computer sciences are discusses, analyzed and evaluated on an international level. The seventh Congress was held in Toronto from August 8 to August 12, 1977, the first time it had been held in North America in 12 years.

The first world computer chess tournament sponsored by IFIP was held at

the 1974 Congress in Stockholm. As part of this year's congress the tournament was again held; it attracted six entries from the United States, three from Canada, two from the U.K. and one each from Germany, the Netherlands, Russia, Sweden and Switzerland.

Computer chess is based upon pre-written computer programs but has one thing in common with regular chess tournaments in that each player (or team) is allowed two hours for a maximum of 40 moves. In computer chess, the program is really the player. The persons operating the terminals act only as a relay between the chess board and the computer and are not allowed to change the program parameters.

The actual computer used is not a significant factor though its capacity and speed of operation could be a factor; obviously the computer of 20 years ago would be unable to match today's equipment.

Why would anyone bother to program a computer to play chess? One reason of course is the sheer intellectual challenge – but there are other justifications.

Techniques used in developing a computer chess program have been used by their authors in writing programs to solve other similar types of problems involving a search among alternative paths. There is also the hope that the development of such programs might provide clues as to how the human brain works to analyze patterns and abstract what is important. Many chess program developers deliberately attempt to simulate human thought processes.

The first comprehensive description of how a computer could be programmed to play chess was given by Claude Shannon, then at Bell Laboratories, in 1949. With the computers then available, Shannon thought it would be possible to look ahead two full moves for each side. Since the number of legal moves available to a player at each turn averages about 30, a look ahead for the two full moves per side would require examination of about 810,000 possible moves. To reduce this, Shannon proposed the elimination of the most obvious of the bad moves. When his plan was implemented in the late 1950's using an IBM 704. The number of discrete moves inspected at each turn had been reduced to 2401. Even at that, a look ahead search of two full moves took eight minutes on the 704.

One team, using a program from McGill University in Montreal had a Data General Supernova on site at the tournament but the other teams used terminals supplied by Bell Canada for time-shared access to large scale computers. Most programs were run on computers in Canada or the U.S. but the West German program was on a Telefunken TR400 computer in Munich and a British entry was on an IBM 270/168 in Harwell, England.

Because of the difficulties and cost of communications between Toronto and Moscow, arrangements were made for the Russian team to use the 370/168 facilities. at Canada Systems Group. Their program had been developed on a Russian built, IBM compatible computer said to be comparable to an IBM 360/50. Before the tournament the Russian team came to the Canada Systems Group data-centre in Mississauga. Here they were assisted by System Engineer Joan Stroud who helped them to become familiar with the operating environment and to put their program up on the system. The program had been written by team members Dr. M.V. Donskoy and Dr. V. Arlazarov at the Institute for Systems Studies in Moscow.

The chess matches were conducted under the rules for a "Swiss style tournament". Basically this means that the winner of each match goes on to meet another winner until only two teams are left for the final match. At this tournament there were sixteen entries, meaning that four rounds were played. After the first round there were eight winners who met in the second round, from the second round four winners emerged, from the third round there were two winners who played off in the fourth and final round. The Toronto tournament was therefore conducted as a "four round Swiss system", with the opponents being ranked and paired by the tournament director.

At the first world tournament in 1974 the Russian team was the winner. They had not met the leading U.S. team from Northwestern University during the tournament so a special match was held. This didn't settle anything because the game ended in a draw.

In this year's competition the winning team was from Northwestern with the Russian team tied for second place with another U.S. team from Duke University. At the end of play another challenge match was held and this time the Northwestern team beat the Russian team.

A program thesis

. . . . . *Mike Valenti*, who with *Prof. Zvonko Vranesic* of the University of Toronto co-authored *Chute 1.2* which participated in the Toronto tournament, has given us permission to reprint his 1974 Master's thesis, "An Easily Modifiable Chess Playing Program," given at the University of Toronto in 1974. Since that time the thesis has been revised several times, and the chess program *Chute 1.2* has its origin in the thesis. Because of its long length the program will be presented in short extracts from time to time. It offers the reader a method of becoming familiar with the complex problems facing chess-program authors:

"*Chute 1*", writes *Mike* in his thesis, "is a chess playing computer-program designed from the outset to be easily modifiable. It has been written in a high level language and implemented on an IBM System 370/165. It employs some new ideas in the look-ahead procedure, where the best path is selected as the best series of moves, rather than the path leading to the best final position, as most other chess-playing programs.

"Einstein once wrote about the combative aspects of chess: 'I have to confess that I have always disliked the fierce, competitive spirit in that highly intellectual game.' On this point, the author and Albert Einstein are in agreement.

"To oversimplify the point, programming is for programmers, and chess playing is for chess lovers. This program was written in close co-opera-

tion with *Dr. Z.G. Vranesic*, an International Master, and any expertise that the program exhibits must be attributed to his chess-specific advice.

"The original intent of the thesis was to provide a tool whereby a person with good ideas on specific chess strategies but little programming experience (and lots of desire) could implement his ideas on a computer. The first attempt at solving the problem was to develop a chess programming language that would allow the user to specify his ideas in chess terms. However, there are many good programmers and many good chess players, but very few good at both programming and chess playing. On this basis, the decision was made to use a high-level programming language and provide a program that is easy to understand and modify for programmers, and let the chess players supply the advice. The chess player needs only to learn the basic concepts of computer chess play, and needn't worry about the nitty-gritties of programming the computer himself.

"The decision was then made to write the basic framework of a chess playing program, keeping in mind that it must be easily modifiable and understandable. Since supplying this framework was the main objective, it was decided that the program should follow the same basic setup as most other successful chess playing programs, but allow the addition of new strategies. During development, however, the temptation arose to fill in much of this framework and incorporate some new ideas in the process. For this reason, the program actually plays a reasonable game of chess under tournament time regulations (40 moves in two hours), although this was not its primary objective. The facility still remains to replace or modify the program's chess strategy.

"The original intention of providing a computer chess strategy exploration tool for a non-programmer is not realized here, but given a programmer-chess enthusiast team (which could be one person or several) many ideas can be investigated using this program as the basis.

"The standard approach used by most successful chess playing programs is to generate a game tree of plausible moves and counter-moves to a certain depth and evaluate the final positions. These terminal positions are assigned values based on such things as centre control, pawn structure, material balance, etc., and their values are backed up to the first level of the tree where the move with the highest value is selected.

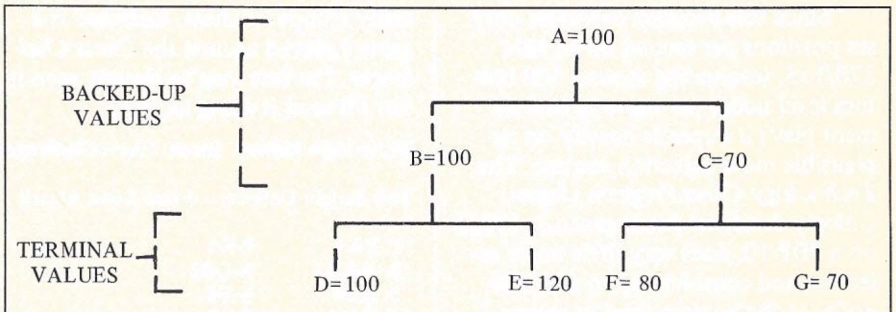


Fig 1 Example of Back-up Procedure.

"The usual back-up procedure is mini-maxing. If a node in the tree represents the computer's choice of move, then the maximum of the values of its successors is selected. If a node represents the opponent's choice, then the minimum value is selected. An example of this mini-maxing is illustrated below. The values are backed up starting from 'D' and 'E' to 'B' to 'A', then 'F' and 'G' to 'C' to 'A', in the following manner.

"The nodes in Figure 1 represent board positions that have been arrived at by some move. For instance, node 'A' is the position arrived at by a move of the computer's opponent. A plausible move selection routine proposes possible responses to the opponent's move and comes up with moves leading to position 'B' and 'C'. The plausible move generator then generates the opponent's responses to these positions and comes up with nodes 'D' through 'G'. Now a static evaluation routine looks at these final positions and gives static scores to these, independent of the plausibility scores used to propose these positions. These scores show how good the final position looks to the computer.

"Now the computer must make a choice of either the move leading to position 'B' or the move leading to position 'C'. From node 'B', the opponent will choose its move to minimize the computer's score, and thus chooses the minimum of 'D' and 'E', namely 100, and this score is 'backed up' to 'B'. This score is then backed up to 'A' for later comparison.

"Similarly, it will make the move leading to position 'G' from 'C' and back up the value 70 to node 'C'. Now, from position 'A', the computer makes the move to optimize its score, namely the move leading to position 'B', and thus the value of 100 is left at 'A'.

"In this example, node 'G' needn't have been evaluated, since 'F' yielding a lower score than 'A' had already backed up from 'D' and 'E'. Optimizing of this sort is referred to as alpha-beta tree pruning.

"The problem with the game of chess is that these trees grow exponentially. A typical position in the middle game (most major pieces on the board) can typically yield 40 legal moves. If this many moves and responses are considered to a reasonable depth of five ply (a ply is one play by one side) then the tree will have 40^5 nodes at the bottom level, which is impossible for even a big computer to evaluate in a reasonable amount of time.

"The way to avoid this is to consider only a fixed number of moves at each level of the tree, occasionally widened with "special" moves. A "special" move is checking, forking or other move that would be nice to look into more deeply.

"The computer must therefore consider only a subset (typically under 10 at each of the first two levels) of the legal moves and throw out the rest. These most plausible moves are then used in creating the tree. The program's calibre of play depends very much on how well it chooses these moves. At present, this program uses 21 heuristics for the plausible move selection. Another factor is the number of positions the program is capable of evaluating in a given amount of time.

"Since this program evaluates only six positions per second on an IBM 370/165, (evaluating around 400 bottom level nodes per move in tournament play) it depends heavily on its plausible move selection routine. The *Technology Chess Program* (James Gillogly Artificial Intelligence 3, 1972) on a PDP-10, does very little move selection and considers up to 500,000 nodes at the bottom level in tournament play. At the other end of the spectrum is the *COKO III* program (E. W. Kozdrowicki and Dennis Cooper, Communications of the ACM Vol. 16, No. 7, 1973) which takes four seconds to evaluate a board on an IBM 360/65, but considers many factors and computes complex exchanges that other programs leave to the look-ahead (tree generation) algorithm.

"A detailed history of chess playing computer programs is contained in *Ray Jones*' thesis (Jones 1971, Dept. of Computer Science, University of Toronto). This program uses Jones' *TUTOR I* chess playing program as a basic reference for the data structure representing a board and its pieces, and the organization of the plausible move selection and heuristics routines. The reasons for choosing Jones' work as the basis are that his program followed the same basic organization as most other successful chess playing programs, his work was done at the University of Toronto and was readily accessible to the author, *Dr. Z.G. Vranesic* was his advisor and is also the author's advisor, and experience gained during its development and use could be applied readily to this work."

(Language-use will be discussed in the next issue.)

Shortest game on record

..... From *Michael K. Molloy*, 5510 17th Street, Lubbock TX 79416:

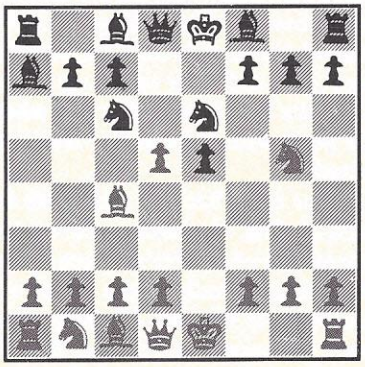
I am a computer technician/maintenance programmer in the Air Force and play "C" strength chess. I've helped write an operating system/compiler for a defense computer. I'm not on that machine anymore, at the moment, but I am interested in developing a chess program for a home computer. I was excited to see *Bobby Fischer*'s letter in *Doug*'s Newsletter. I'm happy to see *Bobby* is still involved in chess, especially computer chess. Enclosed is a game I played against the *Chess Challenger*. I'm sure you've already seen it, but I'll send it along anyway:

White-**Mike Molloy** Black: **Chess Challenger**

Two Knight Defense – Fried Liver Attack

1.	P-K4	P-K4
2.	N-KB3	N-QB3
3.	B-B4	N-B3
4.	N-N5	P-Q4
5.	PxP	NxP

The position after the 5th move was:



6.	NxP	KxN
7.	Q-B3	K-K3
8.	N-B3	N-Q5??
9.	BxN	K-K2
10.	Q-B7	K-Q3
11.	N-K4mate	

This game of 11 moves against **Chess Challenger** appears to be the record for the shortest game played against this program.

LIGHT-PEN used in game

..... From *Andrew Koenig*, 401 Route #46E, North Plainfield, NJ 07060:

If it's of any interest, I'm one of four people who wrote a chess program called *CCCP* which competed in the annual computer chess tournament that was held during ACM '71. The other three are *Steve Bellovin*, now at the University of North Carolina at Chapel Hill, and *Ben Yalow* and *Aron Eisenpres*, both now working at the computer center associated with City University of New York. (*Mr. Yalow*'s name may be a bit familiar: his mother recently won the Nobel Prize . . .) I designed the overall structure of the program and coded much of the human interface (English notation decoder, etc.), *Steve* wrote the tree searching and pruning routines, *Ben* did the move generators and evaluation routines, and *Aron* wrote the part of the human interface that made it possible to enter moves at a 2250 display with a light pen . . . and also rewrote much of my code, substantially improving it in the process. The program was in PL/I, except the graphical interface, which *Aron* wrote in Assembler language. *Ken Thompson* (Bell Laboratories; 600 Mountain Avenue, Murray Hill, NJ 07974) has written a chess program named *Belle* which is unusual in two aspects: it runs on a minicomputer (not so unusual anymore) and it has a hardware generator!.

Bibliography available

..... *Professor Tony Marsland* of the Computing Science Department, University of Alberta, has collected and published one of the best bibliographies on computer chess to date. Almost 300 different titles are listed together with author's name, subject matter and source. The collection lists 80 complete reports or books; 16 theses on computer chess (some of which describe the actual programs currently used in computer chess tournaments) and over 200 articles appearing in various periodicals. The titles include: "Some ideas for a chess compiler," "Tournament Games of a Chess Playing Computer," "Chess Playing Programs," "An introduction to Computer Chess," "Outline for a Strategy for a Chess Playing Program," "A Simple Working Computer Chess Model," and several hundred others. The excellent bibliography can be obtained by requesting "Technical Report TR77-4" from the Computing Science Department, University of Alberta, Edmonton, Alberta, Canada, T6G 2H1 and enclosing $1 to cover the cost. Those who have seen it say it is a valuable addition to the library of either the beginner or advanced participant in chess.

Micro chess tourney

..... The hobby computer chess tournament, held March 3-5, 1978, in San Jose (see accompanying box score), during the West Coast Computer Faire, brought this report from *Doug Penrod*:

"Highlights of the tournament centered on *Steve Stuart* and the *Spracklens*. Steve Stuart's home-brew comput-

er was on a small metal chassis. Enters his stuff in binary via switches, then to verify it, dumps memory in Morse Code, which he listens to while looking at the program. (Octal or hex. Hex I think.) The winning program *Sargon*, was written starting September by *Dan* and *Kathy Spracklen* who didn't get their machine until December! They acquired a ready-made Jupiter II Wave Mate, a 2MHz Z-80 and their program takes less than 8K bytes. After the tournament, *Alan Benson*, local chess master, played all the machines simultaneously, blitz, and he judged the Spracklen program to be the toughest. Eleven of us went to dinner that night, and *Alan* wrote down *all* the blitz games from memory! Saturday night some of us went to the banquet together, including *Timothy Bonham*, of CDC, associated with the *Chess 4.6* people. *Peter Jennings* was at the tournament with a new program for Commodore, but it still has bugs. The Arnstein program *8080 Chess* from the Seattle tournament was there, too, for Processor Technology. Compucolor had a magnificent color display of the chess board for their game. Processor Technology brought three computers along. The marathon participant of the tourney was *Steve Wong* who played 30-35 hours. I noticed that Commodore's notation ranks are numbered backwards. Its play was not as good as expected and could have been due to a bad bit in PROM. Floating around the hall and making themselves useful were *Roy Elder*, *Larry Wagner* and *Walter Korn*. In addition, two local moguls were there to observe the events: *Alan Benson*, chess master and regional VP, USCF; and *John Larkens*, editor of Chess Voice and chess columnist for Berkeley Gazette. The tournament assistants, who all did a fine job, were: *John Keary, Alan Miller* (who fed *Sargon*), *Ian Shepperd, Larry Kaplan, Craig Asher, Brad Stewart, John Mills* and *Daryl Elder. Steve Stuart*'s early victories with his 'metal box' brought lots of spectators swarming in to see the contraption. *Steve* was not defeated when he was playing black. As white, though, he was beaten in the two games he played. On the fourth round *Sargon* and *Chess Mate* agreed to adjourn their lengthy battle so the round could be finished. At the time, *Sargon* had a knight and pawn advantage. The game was concluded before the start of round 5 and *Sargon* emerged the victor. There were a few worried moments when *Sargon* encountered difficulty loading its tape due to a flukey pin plug. We all sweated it out until the problem was finally solved. *Larry Wagner* hopped all over the place making tourney notes, many of which have been incorporated in this report."

Helpful references

. . . . . From *Rolf Sonntag*, Richard Wagner Str. 27, D-3000 Hanover-1, West Germany:

Here are some references which might be helpful to your readers:

G. Veenker, "A Program for Solving Chess Problems" (German) by Elektronische Rechenanlagen 7,1 (1965) 25-29. Describes a program that solves chess problems (mate in two or three moves) by trial and error.

H.W. Wolf, "Program for Solving Chess Problems" (German) Elektronische Datenverarbeitung 7, 1 (1965) 1-14. A simple extension of the program allows to solve the problem "mate after n moves."

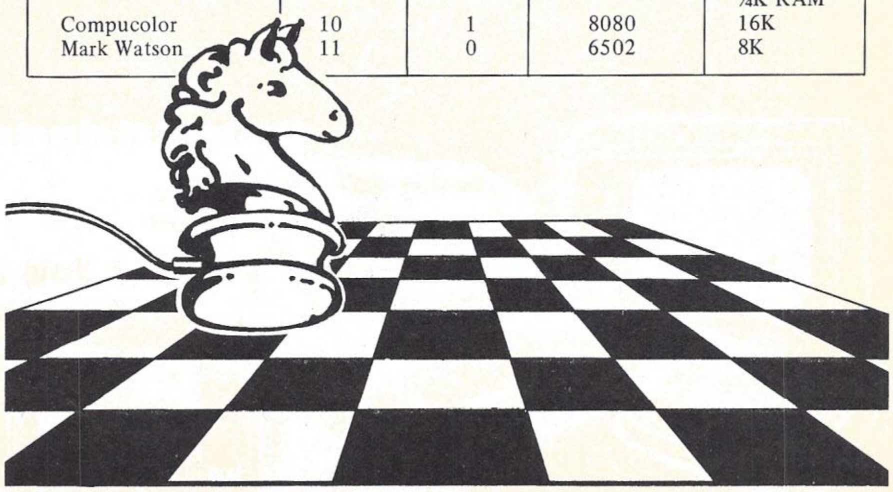
G. Zielinski, "Arrays for Programming Chess." Kybernetes 5 (1976) 91-96. Discusses various representations of the chessboard.

G. Zielinski, "Simple Evolution Functions," Kybernetes 5 (1976) 181-185. The proposed evaluation technique reduces tree searching by introducing arrays of distances and their weights.

R.H. Atkin, W.R. Hartston and I.H. Witten, Fred Champ, "Positional-Chess Analysts," International Journal of Man-Machine Studies 8 (1976) 517-529. A well-defined hierarchical approach is used to produce a vector mapping for the positional evaluation. It is illustrated by an analysis of a grandmaster game, Karpov vs. Spassky.

Microcomputer Tourney In San Jose

Player	Finish	Score (Win=1 Draw=½)	Microprocessor Used	Memory
Sargon	1	5	Z-80	16K
Chess Mate	2 (tie)	3	6504	5K ROM ¼K RAM
Boris	2 (tie)	3	F8	2½K ROM ¼K RAM
Chess Challenger	2 (tie)	3	F8	4K ROM ½K RAM
Processor Technology	5	2½	8080	16K
S D Chess	6 (tie)	2	6800	32K
Tenberg BASIC	6 (tie)	2	F8	?
Steve Stuart	8 (tie)	1½	2650	2K
Compu-Chess	8 (tie)	1½	F8	2K ROM ¼K RAM
Compucolor	10	1	8080	16K
Mark Watson	11	0	6502	8K



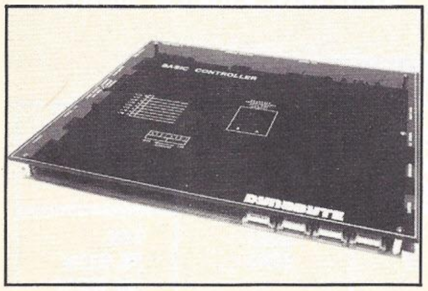
Systems, Subsystems, Software

For company addresses, see Buyer's Guide, p. 124

A single-board programmable microcomputer system designed specifically for control applications was recently introduced by Dynabyte, Inc. The **Basic Controller** allows users to operate their computer and the external devices it controls with a BASIC language called ZIBL.

Dynabyte said it divides the control world into six categories: sense inputs, flag outputs, lites, relays, A/Ds and D/As. ZIBL, the language that the company wrote, implements 64 channels of each category in "such a way," the company claimed, "that the user need not know anything more about them than their names."

According to the company, file structures allow multiple programs written in ZIBL to reside concurrently



in RAM with each program individually Loaded, Renamed or Run. "Any program may access another program as though it were a subroutine, while still retaining its own line numbers and variables," the company said.

Providing communications capabilities, the Basic Controller allows users to List, Print and Input to or from any serial or parallel I/O channel or the self-contained CRT I/O, Dynabyte said.

On-board hardware includes a Z80 MPU, 32 flags, 32 sense, 8 relays, 8 lites, 2 serial I/O, 1 parallel I/O, a cassette I/O, 64 by 16 video I/O, keyboard port, two 2716 sockets with programming capability and up to 16K on-board RAM (4K included). Users interact with the Basic Controller through a keyboard and video monitor that attaches to the printed circuit board. Basic Controllers retail for $750.

Designed for students, scientists, loan officers, executives and engineers, Olivetti's **P6040** is a 17-pound programmable minicomputer and calculator. Priced at $2,300, the system includes a minidisk for program and file storage, 16-column alphanumeric printer, integrated keyboard, 16-character LED display, hardwired Mini BASIC interpreter and 3K of user memory. The unit can prepare, execute and debug programs, perform diagnostic checks on syntax and logic-errors and monitor the workings of its own internal components.

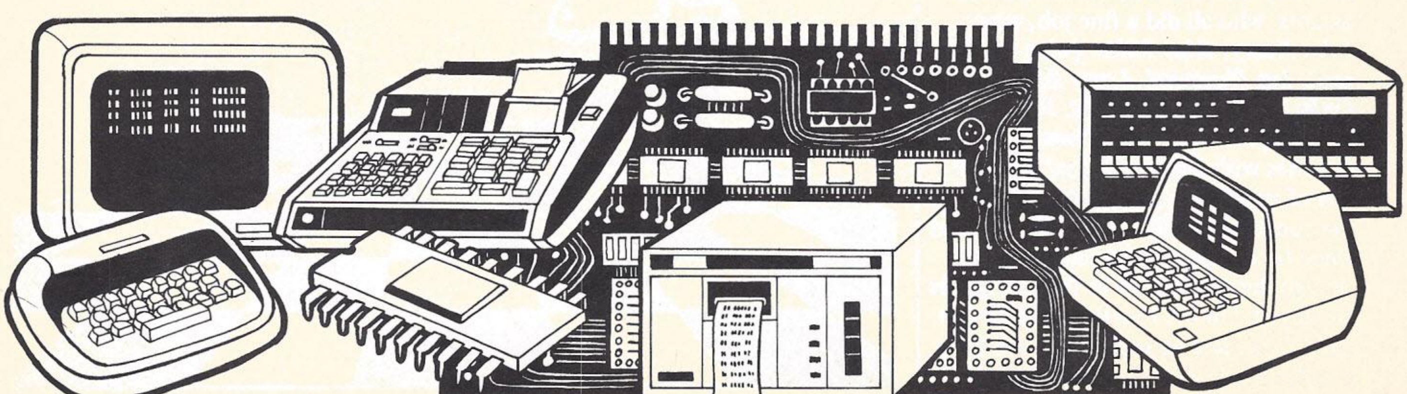
Somewhat larger than a silver dollar and as thin as a sheet of paper, the 2-1/2-inch minidisk provides 3K bytes of storage. P6040 Mini BASIC, a subset of BASIC, is based on seven key programming verbs, located on the keyboard, which also provides 26 user-defined function keys.

Using optional parallel and RS232C serial interfaces, the unit can connect to lab instruments, auxiliary printers, graphic XY plotters, CRTs and external memory devices.

Available software includes calculation of student aid eligibility, auto financing, installment and commercial loans, IRA and Keogh plans and real estate projections.

Two compact office computer systems, **P312** and **P322**, are available from Philips Data Systems. Integration of the floppy disk memories gives the systems higher processing speeds, the manufacturers claim. The computers come with 100-character-per-second printers and can be supplied with 2 or 4 disk drives.

Compatible with any host computer, Sigma Data Systems' **ST-1 Serial Data Translator** performs code translations. The device performs standard translations (ASCII/EBDIC/BAUDOT) as well as terminal control code translations, and the user may program custom applications in Intel MCS-48 compatible machine language.



Let's Get Personal in Anaheim

June 6-8, 1978

A rewarding personal experience is in store for you June 6 - 8 at the NCC '78 Personal Computing Festival...the most comprehensive personal computing event ever held. The Festival, a separate feature of the National Computer Conference, will include approximately 30 program sessions, commercial exhibits of consumer computing products and services, plus a contest featuring individually-designed microprocessor systems and applications. All Festival activities will take place in the Disneyland Hotel Complex, just a few minutes from the Anaheim Convention Center, site of this year's NCC.

Plan now to attend or participate in the big, new NCC '78 Personal Computing Festival. The program will include special paper, panel, and tutorial sessions on such topics as speech synthesis and recognition, computerized music systems, hardware and software design, computer graphics, and small business systems. To assist you in participating, program deadlines have been extended to March 1 for letters of intent covering proposed papers or sessions, and April 1 for submittal of final papers.

The Festival contest provides a unique opportunity to demonstrate your own accomplishments in hardware, software, and applications ranging from home-brew DOS to educational applications and games. Deadline for submittal of information on proposed demonstrations is March 1.

Don't miss the year's most exciting personal computing event. For additional information fill out and return the coupon, or call AFIPS at 201/391-9810.

PC

NCC '78 Personal Computing Festival

c/o AFIPS, 210 Summit Avenue
Montvale, N.J. 07645
telephone: 201/391-9810

☐ Please keep me up-to-date on Festival plans and activities.

☐ Please rush me information on participating in the Festival program sessions and contest.

☐ My company is interested in exhibiting at the Festival.

Name ______________________
Title ______________________
Company ____________ Division ____________
Street ______________________
City ____________ State ______ Zip ______

System 88 is the new line of professional and small business computers from PolyMorphic Systems. Based around an 8080 processor, the hardware includes upper and lower case keyboard with control keys, and video

monitor suitable for full graphics and supports up to 3 mini-floppy drives. Software includes operating system on disk, word processor, BASIC and assembler. BASIC features multi-dimensional strings, numeric arrays, MAT statement, PLOT statement for graphics, program CHAINing, variable cross-referencing, listing by line number, inverse trig and hyperbolic functions, and array functions SUM, PROD, MEAN, STD. Prices for the system (excluding printer) start at $2795.

Extensys Corporation has announced the **EX3000** – a new family of expandable computer systems that provide distributed processing capability for hardware and multi-tasking for system software. The bus-oriented computer contains several processors, some dedicated to a particular function. The host processor assigns tasks to subsidiary processors

Up to eight 9600 baud serial lines can be multiplexed into various peripherals, permitting communication with, for example, 64 remote terminals – each of which consists of a keyboard, CRT screen, two local processors, and another serial line. The remote terminal can also be plugged into the bus as a local operator's terminal.

Software includes EMOS, a multitasking, multi-user operating system providing high-level interface between application programs and hardware components. EMOS provides individual user memory protection and intersystem communication. Extended BASIC, COBOL, and ANSI FORTRAN IV are also provided. Prices range from $10,800 to $148,550.

Infinite Incorporated offers its new microcomputer system, the **UC2000**, in five configurations ranging from empty mainframe card rack to complete system with CPU, memory, multiple floppy disk and printer. The S100-based computer comes with a 12MHz 12-inch CRT, eight-card-slot mainframe, 18A power supply, axial and various keyboard options. Versions B through E are supplied with an 8080-based computer, but any S100-compatible computer can be used in the A version. Prices are: System A with empty mainframe, $995; System

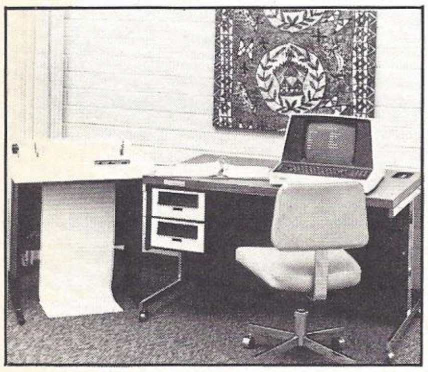
B with CPU, VIF, 8K RAM, Cassette IF, $2266; System C with CPU, VIF, 16K RAM, extended keyboard, one floppy, $3649; System E with features of System C plus RS232 printer IF and dual floppy, $4666.

WHAT'S COMING UP!

Calma offers an erasable video display for interactive graphics featuring a conventional 21-inch video monitor refreshed from a solid state memory. The **Vector Memory Display** (VMD) is offered as an option for all Calma computer-based design, drafting and mapping systems.

With up to four 64-word, 16-bit memory boards, the vector memory holds enough vectors to represent an entire schematic, printed circuit board layout or mask for a complex integrated circuit.

Link-140, a new member of Randal Data Systems' family of business computers, costs $13,353, and can be gradually upgraded into a $55,000 Link-500 as the owner's needs increase. Hardware and software compatible with other Randal systems, the Link-140 features a double-sided, double density floppy disk with 1.2 megabyte



drive. It can be ordered with up to four drives with a total capacity of 4.8 megabytes.

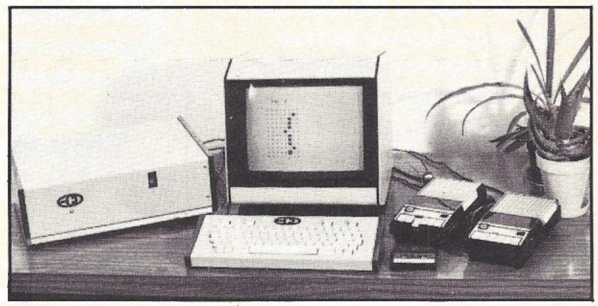
Two people can use the system at the same time with applications such as order entry, inventory control, billing, accounts receivable, accounts payable and payroll. An RTOS I time-sharing operating system uses Business BASIC.

Two 132-column printers are available, with speeds of 30 cps and 70 lpm. The system can have multiple display screens either 12 lines by 80 characters or 24 lines by 80 characters. The typewriter keyboard has upper case characters and a separate numeric pad.

WHAT'S COMING UP!

Texas Instruments' latest product is the **Slimline TI-25**, an 8-digit pocket calculator inside a vinyl wallet with pockets for business cards or note pads. Priced at $30, the three-ounce liquid-crystal-display calculator includes slide-rule functions such as roots, powers, common and natural logarithms, and reciprocals, as well as trig functions, scientific notation memory and pi and factorial keys. Built-in statistical capability computes mean, variance and standard deviation, while algebraic hierarchy allows three levels of parentheses. A "battery saver" feature automatically turns off power after about seven minutes of non-use.

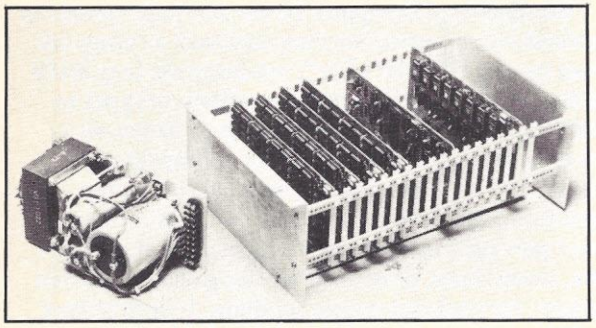
ECD Seven-X microcomputer system allows OEMs to get quickly to market with systems involving previously impossible combinations of features, according to ECD Corp. The display processor handles formats of up to 132 columns. Software can modify display parameters in real-time. Dense text, bold messages and bit-map graphics can be displayed in different windows of the same display. Software available includes a font editor allowing customization of RAM stored fonts.



Applications already in use range from Arabic text editing and real-time star camera displays to computer driven information channels in CATV installations.

The basic system consists of 16K central processor, display processor and general I/O and system support board in an enclosure with power supply and expansion slot. Also furnished are software library, keyboard, universal serial I/O, multi-channel 8-bit A/D and D/A, external I/O bus and display capabilities. Color is an option. Less monitor and mass storage options, the system costs $3900.

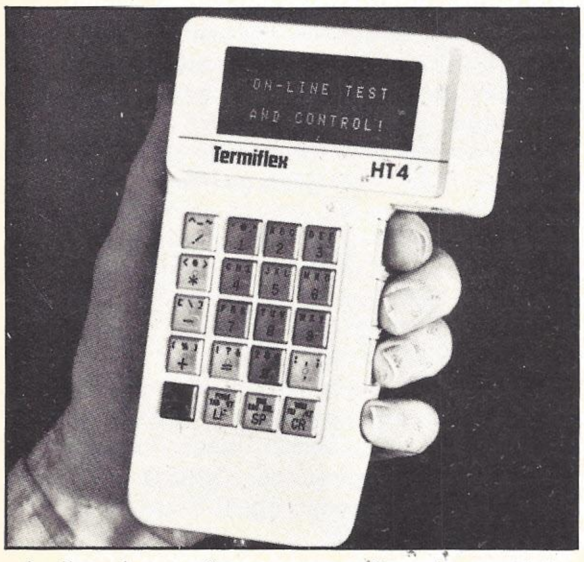
A new I/O board from Ohio Scientific is its **550 16 Port Serial Board.** The board is available in any number of ports from 2 to 16. It is designed to be used on Ohio Scientific Computer systems. The board features RS232 and high speed synchronous interfaces which can be mixed in any combination. The communications transfer rate of each serial port is jumper selectable from a crystal control clock circuit and will support operations from 75 to 19,200 baud synchronous or 250 to 500 Kbits in a synchronous mode. All 16 ports can be jumpered to be continuously addressable memory or can be jumpered to be paged at the same address via Ohio's management hardware. The interface board costs $200 for the first two ports (a CA10-X) plus $50 additional for each extra port up to 16.



Vector 1 computers are now available in a rack mount kit from Vector Graphic. According to the company, the kit includes a card cage, assembled and tested 18-slot motherboard with 18 connectors, card guides and locking buttons for 18 cards. The motherboard is fully shielded to reduce noise on the bus. Price is $225. A companion power supply kit, designed for rack mounting, is also available for $90.

Electric Pencil II, an 8080/Z80 word processor from Michael Shrayer Software, is now available for CP/M. Text is entered and manipulated as a continuous string of characters. Whenever text is inserted or deleted, existing text is pushed down or pulled up in wrap-around fashion. Special features include character spacing, boldface and multicolumn and bidirectional printing (in the Diablo version). The package is $225 for standard printers and $275 for Diablo printers.

Internal rechargeable batteries are now available as a power source for **HT/3** and **HT/4** handheld terminals from Termiflex. According to the company, the terminals allow the user "to compose, edit and transmit messages in remote environments" for applications such as on-line industrial control. Messages may be from 1 to 191 characters long. Price is $300.



Palo Alto Tiny Basic Extended now runs on North Star DOS. The $30 package on diskette is available from California Software and includes string handling functions and SAVE and LOAD commands for disk storage and loading of programs.

WHAT'S COMING UP!

Wordpal from Computer Power and Light produces business letters, assembles long contracts or proposals from standard "boiler plate", produces working drafts of documents requiring several revisions, re-types an "original" letter to each of hundreds of names on a list of clients, prospects, friends or colleagues, and maintains files of documents, lists and commonly used forms, manufacturers claim. Features of the word processor include insertion/deletion of characters, words, lines, or defined blocks; global search and replace; forward and reverse scrolling. Text is printed at 540 words per minute on a daisywheel printer. Printing formats include: variable margins, variable line spacing, right justification, auto centering, underlining, and page numbering. Prices start at $6575 or $171 per month on a 5-year lease/purchase.

Program Utility Package (PUP) for North Star Disk Systems includes five assembler and one BASIC program to enhance the North Star BASIC language. The package includes CREF (produces sorted cross-reference of variable and funcion use by line number), SMPA (produces sorted map of line references by GOSUB and GOTO), FLIST (produces formatted BASIC program listing), MEMTEST (performs multiple memory diagnostics) and HEXMON (loads or lists memory in hexadecimal). Price for the package, which includes six programs on library diskette, 16 pages of documentation and diskette folder, is $25; program listings cost an additional $7.

DEBUG, an on-line program debugger for use with the multi-user MUPRO-80DOS Microprocessor Development System, offers breakpoint, display, modify, control, and trace capabilities. For multi-user configurations, DEBUG provides inter-user protection by allowing each user to prohibit access by his program to areas outside its memory boundaries.

Fast breakpoints allow the program to run at full speed. Up to 10 fast breakpoints may be set, each interrupting upon access of a particular program instruction location. Up to ten emulation breakpoints, independent of and in addition to the fast breakpoints, may be set. Allowable conditions for emulation breakpoints include memory address, range of memory addresses, or type of access.

Display/modification features allow users access to any CPU register or memory location, and selectively display contents in hexadecimal, decimal, octal, binary, or disassembled source code. DEBUG operates on any MUPRO disk system with MUTE release 2.1 and is available for $75.00.

Formscan, a software module for ECRM's 5000 Series Scanners, allows data transmission to a computer for billing, order entry and other applications. Using Formscan eliminates keypunching, editing and verifying usually associated with data entry, resulting in fewer billing errors, manufacturers claim.

The Interdata Division of Perkin-Elmer Corporation has introduced enhanced versions of its **COBOL** and **Indexed Sequential Access Method** (ISAM).

New COBOL features include a high-performance SORT verb, qualification of data names and paragraph names and run-time RETURN CODE. COBOL packaged with ISAM costs $7,000 (US only) with object code distributed on magnetic tape and disk.

The new ISAM can allocate up to 16 memory buffers containing most recently used indexes and data. ISAM utilities allow the user to design, load, and restore optimally arranged file systems. Its run-time routines allow concurrent on-line file use with locking provided to the record level.

ESCOM, Inc., has a **conversion tool** to convert programs written for 2.X systems to 3.0 requirements by changing existing variable names to 3.0 functional names. The conversion runs on either 2.X or 3.0 systems, allowing conversion of programs before or after hardware is converted to 3.0.

From the blue Pacific comes the **Hustler series** of business programs. Computers One of Honolulu offers pre-programmed cassettes (including demo program and instructions) for the Commodore PET (Hustler 1) and the Radio Shack TRS-80 (Hustler 2). Programs available include General Ledger ($19.50), Rent Accounts, Legal Diary and Trust Accounts ($16.95 each). General Ledger and Checking Account require 8K user memory for practical use on the TRS-80. Include $1.50 shipping and handling for each package ordered.

Bored with Star Trek? **"The Devil's Dungeon"** by Engel Enterprises lets you "journey in search of gold and adventure into a bottomless dungeon haunted by fantasmic creatures," according to the developers. The computer game, based on the game "Dungeons and Dragons" by Gygax and Arneson, costs $3.50 ($20 for 10 copies). Engles also offers **"Stimulating Simulations,"** a package of 10 programs, for $5 (10 copies for $30). Included are art auction, monster chase, lost treasure, gone fishing, space flight, forest fire, nautical navigation, business management, rare birds, and diamond thief. Both packages are written in MITS 8K 3.2 BASIC and include scenario, sample run, flowchart, variables listing, program listing and suggested modifications.

A microassembler to aid in microprogramming of bipolar (bit slice) microprocessors is now available from Signetics. The **Micro Assembler** is a software package that can be used for defining microinstructions, writing and assembling programs and generating paper tape output for ROM programming. Written in ANSI FORTRAN IV, the assembler runs on any 16- or 32-bit computer with FORTRAN capability. Cost is $775 in source form on 9-track tape.

TSA Software offers a number of **software packages** for home and small business computers. Software available for Z80/8080/8085 systems include a CPM-compatible operating system ($95), relocatable, linking macro assembler ($95), disk utilities package ($60) and on-screen text editors ($80). TSA also offers TDL, Tarbell and Digital Systems software packages.

Artec Electronics has reduced prices on its 32K byte static memory board, the **32K-100,** from $1055 to $985 both fully assembled and tested. Other-sized boards have been reduced in price also. The Artec company says it is reducing its costs based on anticipated decreases in the price of IC chips. Primary advantage of the Artec 32K-100 is the 24K of memory that can be added to the original 8K card in incremental steps. The board, nicknamed by the company "Expandable Elephant", requires plus-8 single voltage with a power usage of only 3 amps. Access time for the DMA-compatible board is 250 nanoseconds.

A new keyboard, **Model 756,** by George Risk Industries, provides encoding for all 128 ASCII characters and control functions. The new unit, says the company, will bridge the gap between basic keyboards and expensive custom O.E.M. models. Assembly and mounting are simplified through use of industrial grade components and MIL-grade printed circuit boards. Circuitry is complemented by custom molded keytops and heavy-guage steel enclosures for desktop use. The interface allows selection of positive or negative logic data, alpha lock operation and signals from both the D.C. level and pulse strobe. Assembled and tested the model retails for $75.95. The kit is $64.95. A matching enclosure is $29.95.

A new single-board computer, **Model OB8001,** by Omnibyte Corporation, is being marketed by the company. Said to be of potential use in machine control, process control, scientific instrument, military, aerospace, and energy management, and hobby markets, the new low cost CPU module is available without chassis for $237 each in quantities of 100. The single-board computer contains the processor, memory, and I/O on a 4.5" x 6.5" card. It also includes serial communications interface meeting both 20ma current loop and RS-232C standards. The 6800-based processor can be used to implement wide choice of stand-alone controllers.

Creative Microsystems has a new asynchronous serial interface module, **9650.** It is compatible with the M6800 microprocessor bus, the Motorola EXORciser and Micromodules, and with the Evaluation Kit. The 9650 occupies 16 consecutive memory addresses. Lowest 8 of these addresses access the 8 control/status registers and the next 8 access the transmit/receive data registers. This arrangement permits the use of very tight interrupt polling loop. The on-board bit rate generator simultaneously provides 14 standard rates that can be individually strapped to each ACIA. Prices for the 9650 range from $395 in single quantities to $237 at 100.

Texas Instruments has introduced its **SN75363NE,** a new CCD memory driver. The dual-in-line package has six internally connected heat-sink pins for improved heat dissipation. Package leads fit standard 14-pin sockets. It is designed to drive high capacitive loads at frequencies from 1 to 5 MHz. It can also be used to drive the chip-enable clock input, address, control and timing inputs for some MOS RAMs. Price for the memory driver is $3.60 in units of 100.

The **Little Brain I**, by BPI Electronics, is a 6802 based S-11 type single board microcomputer. The board has 8K of UV eraseable PROM and 8K words of static RAM plus an RS-232C channel. Other features include on-board voltage regulators, fully buffered address, data and control buses along with a 128-word scratchpad memory. The fully socketed version with 2K Monitor/Debug program and 1K words of RAM sells for $395 each.

Connecticut microComputers is marketing a new printer adapter, **PET ADA model 1200**, for the Commodore PET. This adapter drives an RS-232 printer from the PET IEEE-488 bus. It allows a PET owner to type letters, manuscripts, mailing labels, data and other tables, using a standard RS-232 printer. The complete PET ADA model 1200 costs $169. It is also available assembled and tested for $98.50 but without power supplies, case or RS-232 connector.

A new, high-resolution **graphics board** has been announced by Vector Graphics. The board is designed to operate in one of two modes: digital output or 16 level gray scale. It requires +8 Vdc and a minimum of 8K RAM memory and will produce digital graphic displays of 256H x 240V screen elements or gray scale 128H x 120V elements. The Video output conforms to RS-170 and will interface to standard raster scale monitors. The new board, which sells for $235 assembled ($195 kit), is designed to be used with Vector Graphic 8K Static RAM memory. Because the board is multiplexed between the bus and the graphics board it can still be used for general purposes when not being used for graphic display.

Computronics Engineering has a new **hexadecimal label keyboard** which is attached to a panel by a self-adhering backing. The micro-file keyboard (.030 thick) avoids the mounting hardware difficulties of mechanical keyboards. A self-contained flex cable plugs into standard 100 mil spacing socket or will allow soldering or wires. The label-keyboard produces matrix coded output directly and will interface with IC 74C922 for binary code. The labels are gold plated and sealed, and its useful life is estimated by the manufacturers to be 100 million operations. After that, for $3.95, you can get a new label keyboard.

The **8085A-2 microprocessor** has been added by Intel Corporation to its series of MCS-85 components. The 5 MHz internal clock of the unit will be able to serve the needs of large system applications as well as those of traditional 8080A and 8085A users, according to the company. The unit improves the ability of the 8085A to better serve the requirements of large system applications. Features include the ability to allow easy decoding of advance status information from the microprocessor; a change in level from 3-state to 2-state for the address latch enable (ALE) signal; and a change in TRAP (non-maskable hardware interrupt) which allows it to be used for general interrupt functions other than catastrophic interrupts to which it was previously restricted. The new 8085A-2 is not yet ready for market, according to Intel, but samples of the item are available.

BUYERS' GUIDE

Basic Controller
Dynabyte, Inc.
4020 Fabian
Palo Alto, CA 94303
Circle No. 101

P6040
Olivetti Corp. of America
Personal Minicomputer Div.
500 Park Ave.
New York, NY 10022
Circle No. 102

P312 and P322
Philips Data Systems
P.O. Box 523
Eindhoven, The Netherlands
Circle No. 103

ST-1 Serial Translator
Sigma Data Systems
715 Torreya Court
Palo Alto, CA 94303
Circle No. 104

System 88
PolyMorphic Systems
460 Ward Drive
Santa Barbara, CA 93111
Circle No. 105

EX3000
Extensys Corp.
380 Bernardo Ave.
Mountain View, CA 94040
Circle No. 106

UC2000
Infinite Inc.
1924 Waverly Place
Melbourne, FL 32901
Circle No. 107

Vector Memory Display
Calma
527 Lakeside Dr.
Sunnyvale, CA 94086
Circle No. 108

Link-140
Randal Data Systems Inc.
365 Maple St.
Torrance, CA 90503
Circle No. 109

Slimline TI-25
Texas Instruments Inc.
Consumer Relations/TI-25
P.O. Box 53
Lubbock, TX 79408
Circle No. 110

ECD Seven-X
ECD Corp.
196 Broadway
Cambridge, MA 02139
Circle No. 111

550 16-Port Serial Board
Ohio Scientific
1333 S. Chillicothe Road
Aurora, OH 44202
Circle No. 112

Vector 1
Vector Graphic, Inc.
790 Hampshire Rd.
Westlake Village, CA 91361
Circle No. 113

Electric Pencil II
Michael Shrayer Software
3901 Los Feliz Boulevard
Los Angeles, CA 90027
Circle No. 114

HT/3 and HT/4
Termiflex Corp.
17 Airport Rd.
Nashua, NH 03060
Circle No. 115

Palo Alto Tiny BASIC Extended
California Software
Box 275
El Cerrito, CA 94530
Circle No. 116

Wordpal
Computer Power and Light
12321 Ventura Blvd.
Studio City, CA 91604
Circle No. 117

Program Utility Package
Business Computer Systems
216 Collier Dr.
Springfield, IL 62704
Circle No. 118

DEBUG
MUPRO
424 Oakmead Parkway
Sunnyvale, CA 94086
Circle No. 119

Formscan
ECRM, Inc.
205 Burlington Rd.
Bedford, MA 01730
Circle No. 120

COBOL and Indexed Sequential Access Method
Interdata Div.
Perkin Elmer Corp.
2 Crescent Place
Oceanport, NJ 07757
Circle No. 121

Conversion Tool
ESCOM
4044 128th SE
Bellevue, WA 98006
Circle No. 122

Hustler Series
Computers One
#306 Kahala Office Tower
4211 Waialae Ave.
Honolulu, HI 96816
Circle No. 123

Software Packages
TSA Software
5 North Salem Rd.
Ridgefield, CT 06877
Circle No. 126

(Continued on page 120)

CLASSIFIEDS

Get the Works, FREE.

"The Complete Bugworks.® From simple solderless breadboards to complete design stations, we have the products, instruments and educational materials (like the famous Bugbook® Volumes) that let you understand and design virtually any circuit system. "The Complete Bugworks" will show you the least expensive, most direct route to mastering the world of electronics. Send for your free copy today.

E&L INSTRUMENTS
61 First Street, Derby, Conn. 06418
(203) 735-8774 Telex No. 96 3536

Please shoot me "The Works."

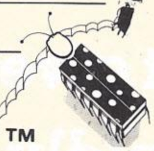
NAME________________

ADDRESS________________

CITY________________

STATE________________

ZIP________________



TM

CIRCLE 41

Z-80 Disassembler

Zilog Mnemonics

North Star of CP/M Disc Systems, Others on special request $30.00/ $35.00 spec.

Microlithics, Inc.
Bits, Bytes & Micros
2918 N. Mac Arthur Blvd.
Oklahoma City, OK 73127
(405) 947-5646

The Small Computer Buying Guide
Volumes 1 and 2 (hardware and software), available 23 Sept.'78.
For more information, address all inquiries to: **The Think Tank**
POBox 1199
Portsmouth, NH 03801

PET OWNERS

Home computing has arrived. To help you get started with a little fun thrown in, try one of our cassettes. Games-1 contains Blackjack, Craps, (plus 6 more you will thoroughly enjoy). Startrek (captain the Enterprise). And to keep your pet warm when you are away, a pet cover.

GAMES-1 (8K)	$15.50
STARTREK (8)	$15.50
PET COVER	$13.75

plus $.75 shipping and handling. Send check payable to:
S.J. White
429 So. Cordova St.
Alhambra, Calif. 91801

Personal Computing Classified Rates

Individual: $1 per word; address set free; first two words set in all capital letters for free. If you want **bold** type, add $1 for each word set bold. Add $5 if you want replies sent to us and forwarded to you.

Company: 1X: $60/inch; 3X: $55/inch; 6X: $50/inch. Display ads accepted only if camera-ready materials submitted. Agency commission paid on camera-ready ads and after the first insertion of a publisher-set ad if subsequent insertions run without changes. Publisher sets ads in format and attempts to fit copy into space requested.

Advertisement accepted, however, with understanding that space may be larger than requested. Minimum billing size one inch; ads billed in half-inch increments thereafter. Frequency rate applies to publisher-set ads only if they run without changes in subsequent issues.

To place an ad, send copy to
Classified Ad Dept.
Personal Computing
1050 Commonwealth Ave.
Boston, MA 02215

BUYERS' GUIDE

32K-100
Artec Electronics, Inc.
605 Old County Road
San Carlos, CA 94070
Circle No. 127

756 Keyboard
George Risk Industries
G.R.I. Plaza
Kimball, NB 69145
Circle No. 128

OB8001 Single Board Computer
Omnibyte Corporation
2711B Curtiss Street
Downers Grove, IL 60515
Circle No. 129

9650 Serial Interface
Creative Micro Systems
6773 Westminster Ave.
Westminster, CA 92683
Circle No. 130

SN75363NE CCD Memory Driver
Texas Instrument, Inc.
Inquiry Answering Service
PO Box 5012 M/S (Attn. SN 75363NE)
Dallas, TX 75222
Circle No. 131

Little Brain I
BPI Electronics, Inc.
4470 S.W. 7th Ave.
Miami, FL 33155
Circle No. 132

Printer Adapter For PET
Connecticut microComputer
150 Pocono Road
Brookfield, CT 06804
Circle No. 133

Graphics Board
Vector Graphic Inc.
790 Hampshire Road
Westlake Village, CA 91361
Circle No. 134

Hexadecimal Label Keyboard
Computronics Engineering
7235 Hollywood Blvd.
Hollywood, CA 90046
Circle No. 135

8085A-2 Microprocessor
Intel Corporation
3065 Bowers Ave.
Santa Clara, CA 95051
Circle No. 136

Devil's Dungeon and Stimulating Simulations
Engel Enterprises
P.O. Box 16612
Tampa, FL 33687
Circle No. 124

Micro Assembler
Signetics
P.O. Box 9052
811 East Arques Ave.
Sunnyvale, CA 94086
Circle No. 125

Big Brother... Sooner than you think?

BY H. PARIS BURSTYN

In Japan and Great Britain cable television provides viewers with two-way communication services. Viewers can receive broadcast information as well as transmit data to broadcasters, other viewers or to a central computer through their television via push-button terminals.

Japan's system also provides flash newscasts – election results, flood warnings and other important and timely events broadcast from remote locations right into the viewer's home.

But without careful monitoring these systems, as advantageous as they presently appear, could develop into Big Brother systems *à la* Orwell's *1984.*

Based on a cable television broadcasting system, Japan's Coaxial Cable Information System (CCIS) incorporates an "automatic telecasting service" within its two-way transmission system. This service places two televisions – one color set and one black-and-white set in one unit – in each home. Information broadcast to these auxiliary televisions appears "automatically" – in other words, without the owner ever turning the set on.

"Public safety" provides the *raison d'etre* for this service, which was designed to play a key role in sending urgent information – news of a major fire or earthquake.

In England, an experimental system called Viewdata provides a kind of televised encyclopedia. The system uses a modified television set linked to an ordinary telephone to enable viewers to select information from a large data base housed in a central computer. Information ranges from up-to-the-minute news to household hints and jobs, travel and vacation data.

To get Viewdata information, viewers turn on their television and call the Viewdata center by pressing a button on the specially provided push-button unit. Viewers do not even have to lift the telephone receiver. In effect, the television becomes an extension phone.

Think about this system together with the Japanese system – modified televisions capable of two-way data flow and the capability of being turned on and off by someone other than you.

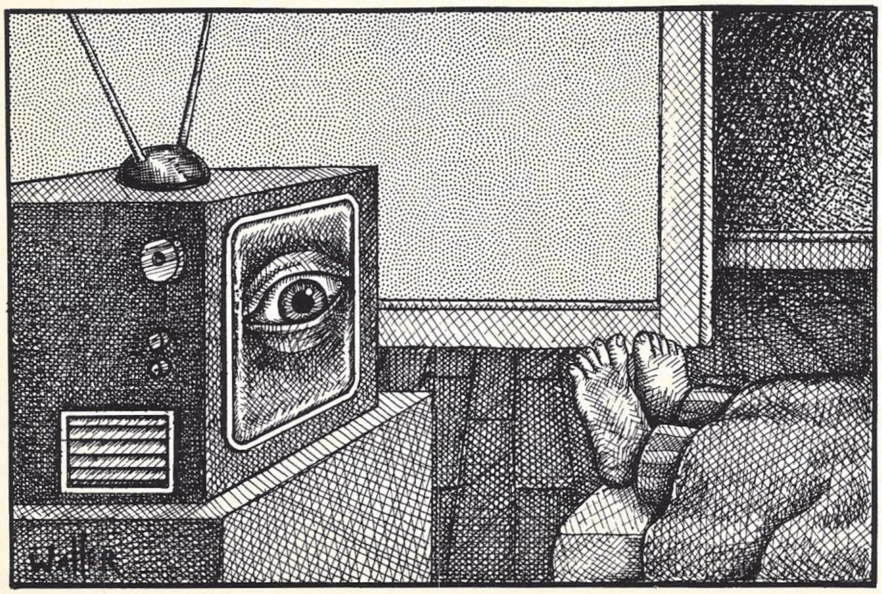


Illustration by Charles Waller

Does your home computer system include a CRT? Basically a CRT is a modified television; couldn't CRTs incorporate remote control capability? Could others gain access to your computer – and its data – through your CRT? The technology exists today.

Just how private is your data? How secure is it from change?

CCIS and Viewdata exist only in experimental form in foreign countries – but technological imports are a matter of daily occurrence.

Not too distant technological advances – the Bell System already employs Picturephones in their business offices – could provide image as well as data transmission from homes in the United States via telephone/television lines.

With marginal public awareness, the United States Federal Communications Commission required cable television companies to install two-way (coaxial) cables. And so, U.S. cable television sets *already* hold a send/receive capability. All they need is importation of the remote control capabilities available in Japan. Or do they?

Remember those beautiful Martian landscapes Viking sent back to Earth? Those television cameras were turned on from Earth. No little green beings obliged NASA by turning on the camera.

United States cable television could have remote turn on and two-way transmission facilities tomorrow.

We have a history of illegal wire taps and eavesdropping in the highest national offices. With available technology and a corrupt political administration, *1984*'s Big Brother could be less than 6 years away.

While two-way cable and remote turn-on can provide convenient and even life-saving services, these facilities also invite abuse and misuse of technology. In such situations, the public interest most often lies in careful and close monitoring of further developments and applications of such systems.

H. Paris Burstyn is an associate editor of Minicomputer News *who frequently writes about the electronics and computer industries.*

ADVERTISERS' INDEX

SALES OFFICES

Northeast: George Palken, 1050 Commonwealth Ave., Boston, MA 02215; (617) 232-5470 • **Mid-Atlantic**: Arthur Daks, 104 So. Central Ave., Valley Stream, LI, NY 11580; (516) 872-9550 • **Mid-West**: Ralph Petersen, 1 Wheaton Ctr. No. 1706, Wheaton, IL 60187; (312) 653-2480 • **Southeast**: Harold Short, Parker & Short, 1 Lakeside Office Park, Wakefield, MA 01880; (617) 246-2293 • **West**: Yuri Spiro & Carol Stagg, 924 Westwood Blvd., Suite 610, Los Angeles, CA 90024; (213) 478-3017 • **Japan**: Hiro H. Irie, International Business Corp., 11-8, Narita-Higashi 1-chome, Suginami-KU, Tokyo 166; Phone: (03) 311-1746.

Now, a complete business computing system for only $5995

Introducing ABACUS 1, An Easy-To-Use Accounting And Word Processing System Here's a total business system you can count on for your general ledger accounting, accounts receivable, accounts payable, payroll, inventory, mailing lists and word processing. ABACUS 1, developed by Computer Products of America, is simple to use and features a powerful software package that can typically handle up to 700 customer listings, 1000 inventory line items, 800 vendors, 300 employees and 60 general ledger accounts. In addition, our word processing option features a character-oriented system for your everyday business communications needs.

The ABACUS 1 is an interactive, double-entry bookkeeping system. It is a complete hardware and software package which includes a Z-80 microprocessor, a dual Northstar diskette system, a video display with a standard keyboard, and a printer. It operates using BASIC which is the easiest programming language to use. And, our attractive design blends with any office decor.

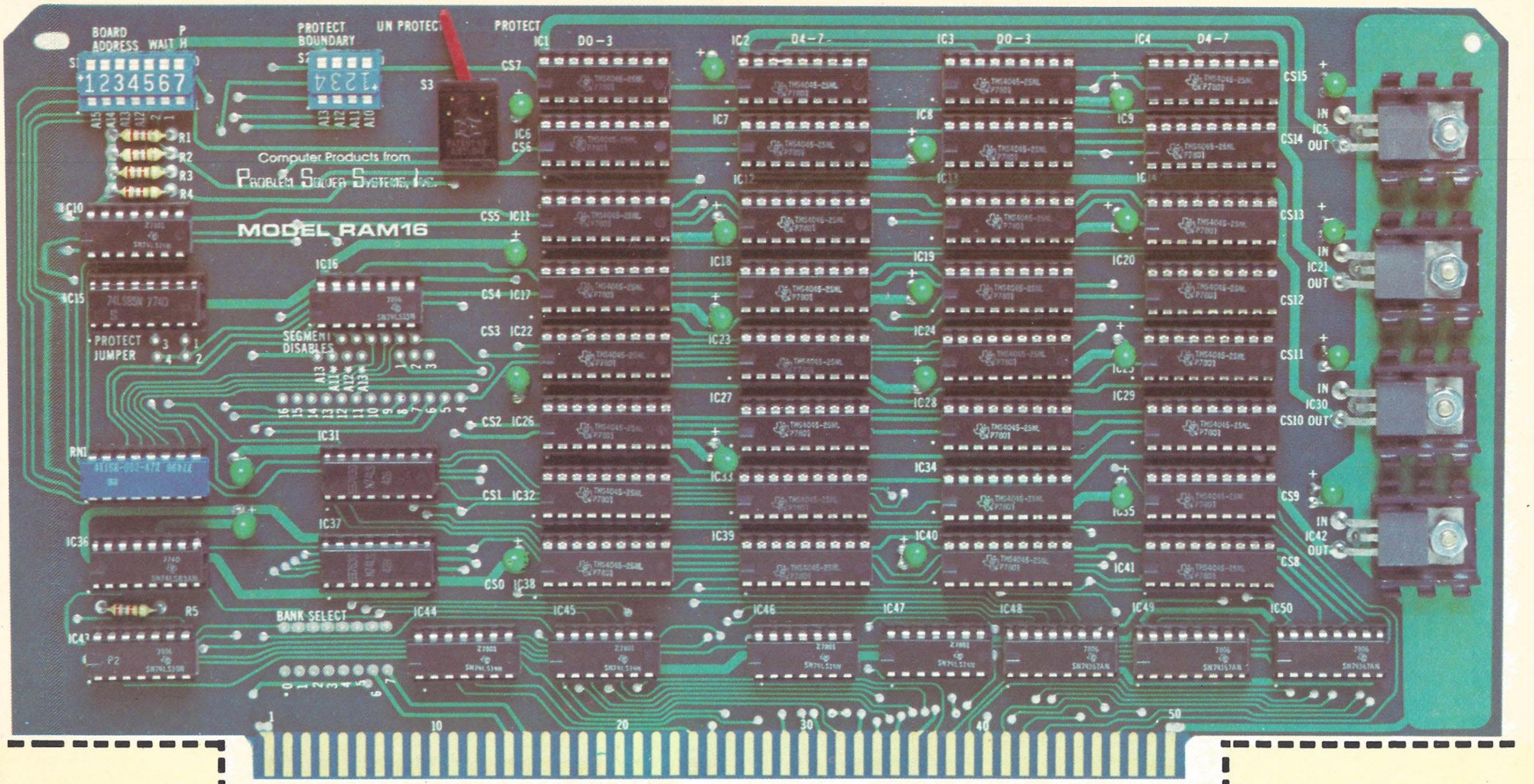
No matter what business you are in, the ABACUS 1 works for you. It's flexible so you can easily change programs to meet your specific requirements. It's backed by our fully staffed service department. And, it's priced to fit your budget. For a business system you can count on, write or stop in today. ABACUS 1 —The Business System You Can Count On.

THE COMPUTER MART

633 West Katella Avenue, Orange, California 92667
(714) 633-1222
Hours: Tuesday–Friday 12:00–8:00
Saturday 9:00–6:00

See ABACUS 1 AT NCC
Visit our Booth #38. Or visit our store. Only five minutes from NCC. Open during NCC every night until 9:00.

CIRCLE 4


Computer Products from
Problem Solver Systems, Inc.
MODEL RAM16
BOARD ADDRESS
PROTECT BOUNDARY
UN PROTECT
PROTECT
PROTECT JUMPER
SEGMENT DISABLES
BANK SELECT